Tampereen ammattikorkeakoulu

Security Onion tunkeutumisen havaitsemisjärjestelmän käyttöönotto

Juha-Matti Karppi

OPINNÄYTETYÖ
Toukokuu 2020

Tietojenkäsittely
Tietoverkkopalvelut

TIIVISTELMÄ

Tampereen ammattikorkeakoulu
Tietojenkäsittely
Tietoverkkopalvelut

KARPPI, JUHA-MATTI:
Security Onion tunkeutumisen havaitsemisjärjestelmän käyttöönotto
Opinnäytetyö 51 sivua
Toukokuu 2020

Opinnäytetyön toimeksiantajana toimi Tampereen ammattikorkeakoulun tietojenkäsittelyn koulutusohjelma. Tavoitteena oli lisätä ammattikorkeakoulun verkon sisällä toimivan WPK-verkon tietoturvallisuutta parantamalla verkon ylläpidon havainnointikykyä. Tätä tarkoitusta varten verkkoon asennettiin tunkeutumisen havaitsemisjärjestelmä ja verkonvalvonta-alusta Security Onion, joka mahdollistaa verkon liikenteen tarkkailun, ja vaarallisen liikenteen tunnistamisen, uhista hälyttämisen sekä liikenteen visualisoinnin. Järjestelmää on tarkoitus käyttää myös opetuksen tukena.

Työssä selvitetään verkon valvonnan merkitystä ja siihen käytettäviä työkaluja, sekä kuvataan niiden toimintaa Security Onion -järjestelmässä. Työn lopputuloksena verkkoon asennettiin Security Onionin versio 16.04.6.2, joka toimii verkossa omalla palvelimellaan. Siitä luotiin itsenäisesti toimiva järjestelmä, joka mahdollistaa WPK-verkon ylläpidolle varsin tarkan ja yksityiskohtaisen verkon valvonnan. Järjestelmää on mahdollista käyttää myös etäyhteydellä.

Työn tuloksia ja materiaaleja voidaan hyödyntää sekä toimeksiantajan käyttötarkoituksissa että myös muissa vastaavissa kohteissa. Työtä on mahdollista kehittää ja muokata toimeksiantajan tarpeiden mukaan.

Asiasanat: ids, tunkeutumisen havaitsemisjärjestelmä, security onion, tietoturva, käyttöönotto

ABSTRACT

Tampereen ammattikorkeakoulu
Tampere University of Applied Sciences
Degree programme in Business Information Systems
Network Services

KARPPI, JUHA-MATTI:
Implementing Security Onion Intrusion Detection System

Bachelor's thesis 51 pages
May 2020

The thesis was commissioned by the Tampere University of Applied Sciences' Degree programme in Business Information Systems. The purpose of the thesis was to increase the information security of the “WPK-verkko” intranet operating within the University of Applied Sciences' network by improving the detection capability of the network administration. For this purpose, an intruder detection system and a Network Security Monitoring platform Security Onion was installed in the network, which enables monitoring of network traffic, detection of dangerous traffic, alerting of threats and visualization of traffic. The system is also meant to be used as a part of teaching.

The thesis examines the importance of network monitoring and the tools used for it and describes their operation in the Security Onion system. As a result of the thesis, Security Onion version 16.04.6.2 was installed on the network, on a a dedicated server. It operates independently and provides the WPK-network administrators precise and detailed network monitoring. It can also be used via remote access.

The results and the materials of the thesis can be utilized both for the client's uses and in other similar applications. It enables further development and modification.

Key words: ids, intrusion detection, security onion, network security, implementation

SISÄLLYS

LYHENTEET JA TERMIT

IDS	Indtrusion Detection System, tunkeutumisen havaitsemisjärjestelmä
HIDS	Host-based Indtrusion Detection System, konekohtainen tunkeutumisen havaitsemisjärjestelmä
NIDS	Network-based Indtrusion Detection System, verkkopohjainen tunkeutumisen havaitsemisjärjestelmä
Pcap	Pakettien kaappauksessa käytetty tiedostomuoto
RDP	Remote Desktop Protocol, Microsoftin kehittämä etäyhteysprotokolla

1 JOHDANTO

Yritysten ja järjestöjen toiminta on nykyisin suurelta osin ja yhä enenevässä määrin riippuvaista verkossa toimivista järjestelmistä. Siksi myös näiden järjestelmien suojaaminen tietoturvauhilta ja hyökkäyksiltä on entistä tärkeämpää. Kyberrikollisuus on yksi nopeimmin kasvavia rikollisuuden muotoja ja aiheuttaa valtavia haittoja niin yrityksille, järjestöille kuin yksityisille käyttäjille, sekä korjauksina että myös toiminnan keskeytymisenä. Tämän vuoksi uhkiin varautuminen, niiden havaitseminen ja ehkäiseminen on ensisijaisen tärkeää. Passiivisen tietoturvan, kuten pääsylistojen, palomuurien ja lokien keräämisen lisäksi tarvitaan myös aktiivisempia keinoja suojata järjestelmiä. Tällaisia keinoja tarjoavat tunkeutumisen havaitsemis- ja estämisjärjestelmät.

Tämän opinnäytetyön tavoitteena on parantaa Tampereen ammattikorkeakoulun sisällä toimivan tietojenkäsittelyn koulutusohjelman WPK-verkon ylläpidon havainnointikykyä. Tällä tarkoitetaan sitä, miten verkon ylläpito saa tiedon verkossa tapahtuvista poikkeustilanteista. Opinnäytetyön tarkoitus on lisätä kykyä vastata mahdollisiin tietoturvauhkiin. Tätä tarkoitusta varten verkkoon asennetaan käyttöön tunkeutumisen havaitsemisjärjestelmä, Security Onion, joka tarkkailee verkon liikennettä ja tuottaa siitä erilaisia raportteja. Lisäksi Security Onion mahdollistaa paremman tilannekuvan verkon liikenteestä.

Opinnäytetyö koostuu teoriaosuudesta sekä käytännön toteutuksesta. Teoriaosuudessa kuvataan verkon valvonnan merkitystä sekä tässä työssä käytettäviä työkaluja. Käytännön työstä esitetään kuvaus sen toteutuksesta sekä käydään läpi sen toiminnallisuuksia. Opinnäytetyön tuotosten, ensisijaisesti Security Onion -järjestelmän on tarkoitus jäädä WPK-verkon ylläpidossa toimivien harjoittelijoiden käyttöön. Sen avulla he pystyvät havainnoimaan verkkoa ja havaitsemaan siinä ilmeneviä tietoturvauhkia ja reagoimaan niihin. Lisäksi pohditaan järjestelmän jatkokehittämistä verkon ylläpidon tarpeisiin sekä arvioidaan työn hyödyllisyyttä.

2 VERKON VALVONNAN MERKITYS

Ongelmat suorituskyvyssä ja verkkoliikenteen katkokset eivät ole ainoastaan turhauttavia korjata, vaan ne tulevat myös erittäin kalliiksi (Hein 2019). Kyberrikollisuuden arvioidaan aiheuttavan yrityksille jopa kuuden biljoonan dollarin kulut vuoteen 2021 mennessä. Tietoverkkorikollisuus voi tarkoittaa muun muassa immateriaalioikeuksien varkauksia, tietojen tuhoamista, varastettua rahaa, kavalluksia ja se voi vähentää tuottavuutta, tuottaa hyökkäyksen jälkeisiä yritystoiminnan häiriöitä ja maineeseen kohdistuvia haittoja. Koska yritysten toiminta on erittäin riippuvaista digitaalisista järjestelmistä, on tunkeilijoilta ja hyökkäyksiltä suojautuminen nykyisin välttämätöntä. (MHC Datacomm Inc 2019.)

Yksinkertaisin valvontapa on loki- ja tilastotietojen seuraaminen, mutta pienessäkin verkossa tietoja kertyy todella paljon. Niiden läpikäyminen vaatisi yläpitäjältä aikaa vievää mekaanista työtä, joten se on kannattavampaa automatisoida. Yksi ratkaisu hyökkäysten tai muun epätavallisen toiminnan havaitsemiseen on tunkeutumisen havaitsemisjärjestelmä eli IDS, intrusion detection system. (Mäntylahti 2003.)

Tunkeutumisen havaitsemisjärjestelmän päätehtävä verkossa on auttaa tietokonejärjestelmiä valmistautumaan ja käsittelemään verkkohyökkäyksiä (Ashoor & Gore n.d). Palomuurin ja sen suojaaman järjestelmän väliin asetettuna tunkeutumisen havaitsemisjärjestelmä tarjoaa yhden turvallisuuden tason. Sen avulla voidaan esimerkiksi tarkkailla, onko palomuuria murrettu tai onko sen ohittamiseen käytetty aiemmin tuntematonta mekanismia. (Elson 2000.)

Richard Bejtlichin (2013) mukaan turvamurrot ovat yrityksissä ja organisaatioissa vääjäämättömiä ja kaikki ehkäisy pettää lopulta. Tunkeutumisen havaitsemisjärjestelmien ja verkon valvonnan tarkoitus onkin turhauttaa hyökkääjiä ja murron sattuessa estää hyökkääjää saavuttamasta päämääräänsä.

3 TUNKEUTUMISEN HAVAITSEMISJÄRJESTELMÄT

Tunkeutumisen havaitsemisjärjestelmä (IDS) on yksittäisten tietokoneiden tai verkkojen valvontaan tarkoitettu turvallisuusjärjestelmä. Järjestelmä kerää tietoa useista erilaisista lähteistä tunnistaakseen mahdolliset tietoturvariskit. Sen avulla voidaan tarkkailla tietokoneilla tai tietoverkoissa tapahtuvaa verkkoliikennettä ja havaita mahdolliset organisaation ulkopuolelta tulevat tunkeilijat, hyökkäysyritykset sekä organisaation sisäiset väärinkäytökset ja muut tietoturvariskit. (Gupta ym 2017.)

Tom Thomasin (2005, 322) mukaan tunkeutumisen havaitseminen perustuu kolmeen edellytykseen. Ensimmäisenä edellytyksenä on missä vahditaan, eli mihin IDS sijoitetaan. Toinen edellytys, mitä vahditaan, määrittää millaisissa tilanteissa IDS:n tulee tehdä hälytys tai suoritetaan jokin muu toimenpide. Kolmantena edellytyksenä on IDS:n toiminta tietyt odotusarvot täyttävässä tilanteessa, eli miten toimitaan.

IDS toimii palomuurin rinnalla täydentäen sitä. Palomuuri estää vahingollisia hyökkäyksiä, mutta se voi olla mahdollista murtaa tai ohittaa. IDS havaitsee tällaiset tilanteet ja lähettää ylläpitäjälle hälytyksen mahdollisesti haitallisista tapahtumista. (Sarmah 2001.)

IDS:n avulla voidaan seurata muutoksia verkon toiminnassa, tarkastella järjestelmän aktiivisuutta, erottaa normaali ja epänormaali toiminta verkossa automatisoidusti. IDS:n ongelmana voidaan pitää sitä, että sen ylläpito vaatii kuitenkin jatkuvaa seurantaa, sillä usein saattaa syntyä myös aiheettomia hälytyksiä, eli ns. false positiveja, mikä on hyvin aikaa vievää. IDS ei myöskään takaa täydellistä suojaa hyökkäyksiltä. (Gupta ym 2017.)

IDS:t voidaan jakaa ryhmiin sen mukaan tarkkailevatko ne yhtä konetta vai koko verkkoa sekä niiden käyttämän haitallisen liikenteen havaitsemistavan mukaan. Tämän opinnäytetyön aiheena oleva Security Onion sisältää kaikkien näiden ryhmien mukaiset IDS:t (Security Onion Solutions_a n.d).

Yksittäistä konetta tarkkailevaa järjestelmää kutsutaan konekohtaiseksi eli Host based IDS:ksi ja verkkoa tarkkailevaa verkkopohjaiseksi eli Network based IDS:ksi (Sarmah, 2001). Lisäksi olemassa on näiden järjestelmien ominaisuuksia yhdisteleviä hybridijärjestelmiä (Ashoor & Gore n.d).

Havaitsemistavan mukaan IDS:t jaetaan signature-based/rule-driven -tyyppisiin ja anomaly-based/analysis-driven -tyyppisiin järjestelmiin. Näistä ensimmäisen havaitsemismetodi perustuu ennalta määritettyihin sääntöihin ja jälkimmäisen liikenteen analysointiin. (Ashoor & Gore n.d.) Näiden peruskategorioiden lisäksi IDS:t voidaan jakaa alakategorioihin (taulukko 1) niiden käyttämien mekanismien ja protokollien mukaan (Axelsson 2000).

TAULUKKO 1. IDS-järjestelmien kategoriat

Table 1: Classification of detection principles

anomaly	self-learning	non time series	rule modelling	W&S A.4[a]
			descriptive statistics	IDES A.3, NIDES A.14, EMERALD A.19, JiNao A.18, Haystack A.1
		time series	ANN	Hyperview(1)[b] A.8
	programmed	descriptive stat	simple stat	MIDAS(1) A.2, NADIR(1) A.7, Haystack(1)
			simple rule-based	NSM A.6
			threshold	ComputerWatch A.5
		default deny	state series modelling	DPEM A.12, JANUS A.17, Bro A.20
signature	programmed	state-modelling	state-transition	USTAT A.11
			petri-net	IDIOT A.13
		expert-system	NIDES A.14, EMERALD A.19, MIDAS-direct A.2, DIDS A.9, MIDAS(2) A.2	
		string-matching	NSM A.6	
		simple rule-based	NADIR A.7, NADIR(2) A.7, ASAX A.10, Bro A.20, JiNao A.18, Haystack(2) A.1	
signature inspired	self-learning	automatic feature sel	Ripper A.21	

[a] Letter and number provide reference to section where it is described [b] Number in brackets indicates level of two tiered detectors.

3.1 Havaitsemismetodit

Epäilyttävän liikenteen havaitsemiseen voidaan käyttää kahta erilaista metodia, ennalta määritettyihin sääntöihin tai liikenteen analysointiin perustuvaa. (Ashoor & Gore n.d.)

3.1.1 Sääntöihin perustuva

Tämä havaitsemistapa on tehokas erityisesti tunnettuja hyökkäyksiä vastaan. Sillä ei kuitenkaan ole mahdollista havaita aiemmin tuntemattomia uhkia tai niiden uusia versioita. Siksi se on riippuvainen säännöllisistä päivityksistä. (Ashoor & Gore n.d.)

3.1.2 Liikenteen analysointiin perustuva

Liikenteen analysoinnin perusteella havaitseminen perustuu verkon toiminnan ja liikenteen normaalin toiminnan ja siitä poikkeavan tilanteen määrittämiseen. Se kykenee havaitsemaan myös aiemmin tuntemattomia uhkia, mutta se voi myös tuottaa enemmän vääriä hälytyksiä. (Ashoor & Gore n.d.)

Tämän tyyppiset järjestelmät voidaan vielä jakaa tarkemmin itseoppiviin ja ohjelmoituihin. Itseoppivat järjestelmät oppivat itsenäisesti mikä on normaalia, tyypillisesti tarkkailemalla verkon liikennettä ja rakentamalla sen perusteella mallin. Ohjelmoituihin järjestelmiin normaalitila ja siitä poikkeavat tilanteet täytyy määrittää. (Axelsson 2000.)

3.2 NIDS – verkkopohjainen havainnointijärjestelmä

Network-based Intrusion Detection System eli verkkopohjainen tunkeutumisen havaitsemisjärjestelmä valvoo ja analysoi verkon liikennettä tietoturvauhkien, kuten porttiskannausten, palvelunesto- ja muiden hyökkäysten varalta. Se lukee kaikki sille ohjatut paketit ja etsii niistä epäilyttäviä malleja. Kun uhkia havaitaan, järjestelmä voi reagoida niihin vakavuuden mukaan eri tavoin, kuten ilmoittamalla niistä ylläpitäjälle tai estämällä lähde-IP:n pääsyn verkkoon. (Techopedia 2011.)

Toiminnaltaan NIDS on samankaltaista kuin ns. pakettien nuuskiminen, mutta kaapatut paketit käsitellään eri tavalla. Kaapatusta liikenteestä luettuja paketteja NIDS analysoidaan joko vertaamalla pakettia tietokannassa oleviin hyökkäystunnusmerkkeihin tai tunnistamalla epätavallista liikennettä. NIDS-järjestelmien etuna on, että tämä tapahtuu käyttäjältä huomaamattomasti. (Thomas 2005, 330-331.)

NIDS:n kytkentä verkkoon voidaan toteuttaa joko lisäämällä verkkokaapeliin haaran johon järjestelmä kytketään tai peilaamalla kytkimeltä liikenne järjestelmälle. Toimiakseen tehokkaasti IDS:n on nähtävä mahdollisimman suuri osa verkon lii-

kenteestä. Verkon muoto ja arkkitehtuuri määrittää myös NIDS-järjestelmien käytön, määrän ja sijoittelun. Perinteisesti NIDS-järjestelmät on sijoitettu verkon reunoille, mutta on suositeltavaa harkita sekä sisäisen että ulkoisen järjestelmän käyttöä. (Thomas 2005, 330-331.)

3.3 HIDS – konekohtainen havainnointijärjestelmä

Host-based intrusion detection system eli konekohtainen tunkeutumisen havaitsemisjärjestelmä asennetaan yksittäiselle isäntäkoneelle, kuten palvelimelle tai työasemalle. Se tarkkailee ja analysoi isäntäkoneensa toimintaa turvallisuuskäytäntöjen sisäisen tai ulkoisen rikkomisen varalta. Havaitessaan mahdollisen vahingollisen tapahtuman, kuten tunkeutumisen ja/tai väärinkäytöksen se reagoi niihin tallentamalla niistä tiedon lokiin ja luomalla hälytyksen ylläpitäjälle. (Techopedia 2017.)

Se soveltuu organisaation toiminnan kannalta välttämättömien, mutta haavoittuvien palveluiden, kuten www-, posti- ja muiden julkiseen Internetiin yhdistettyjen sovelluspalvelimien suojaamiseen. HIDS toimii virustorjunnan tavoin, mutta ei korvaa sitä, vaan toimii sen rinnalla ja sen laajemmat ominaisuudet voivat parantaa tietoturvaa merkittävästi. (Thomas 2005, 331-332.)

Pakettien sijaan se tarkkailee isäntäkoneensa tarkastus- ja tapahtumalokeja ja pyrkii tunnistamaan niistä merkkejä paikallisen tai etäkäyttäjän kielletystä toiminnasta. HIDS vahtii tiedostojärjestelmän ja rekisterin muutoksia, avoimia portteja, käytettäviä sovelluksia sekä isännälle saapuvaa ja siltä lähtevää liikennettä. Sillä voidaan tunnistaa tunkeutumisyritykset, onnistuneet tunkeutumiset ja valtuutettujen käyttäjien epäilyttävä käyttäytyminen. (Thomas 2005, 331-332.)

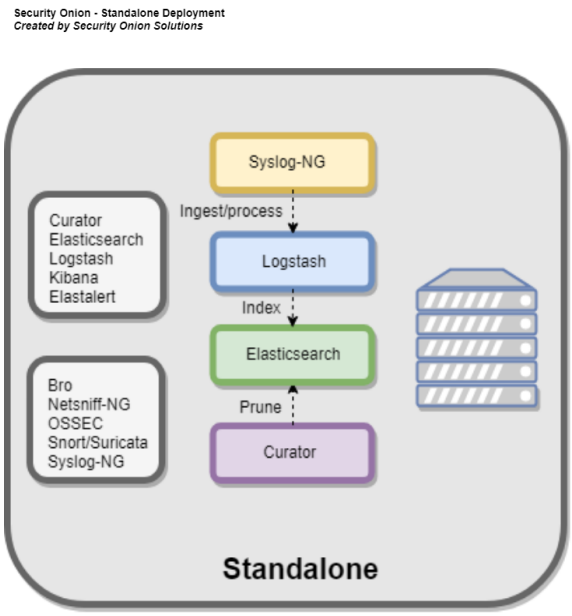
Lisäksi se mahdollistaa tunkeutumisen havaitsemisen ohella muita etuja kuten Isäntäkoneen skannauksen, jolla voidaan varmistaa tietoturvakäytäntöjen noudattaminen. Sen avulla voidaan hallita ja keskittää tarkastuskäytäntöjä, kerätä todistusaineiston rikkomuksista, analysoida kerättyä tietoa ja joissain ohjelmistoissa sillä voidaan suorittaa myös pääsynvalvontaa. (Thomas 2005, 331-332.)

4 SECURITY ONION

Tässä kappaleessa kuvataan tarkemmin opinnäytetyön aiheena olevan Security Onionin rakennetta ja siihen kuuluvia osia ja työkaluja. Security Onion on Doug Burksin vuonna 2008 aloittama avoimen lähdekoodin projekti, jonka hallinnointia varten tämä perusti vuonna 2014 Security Onion Solutions LLC:n (Security Onion Solutions_b n.d.).

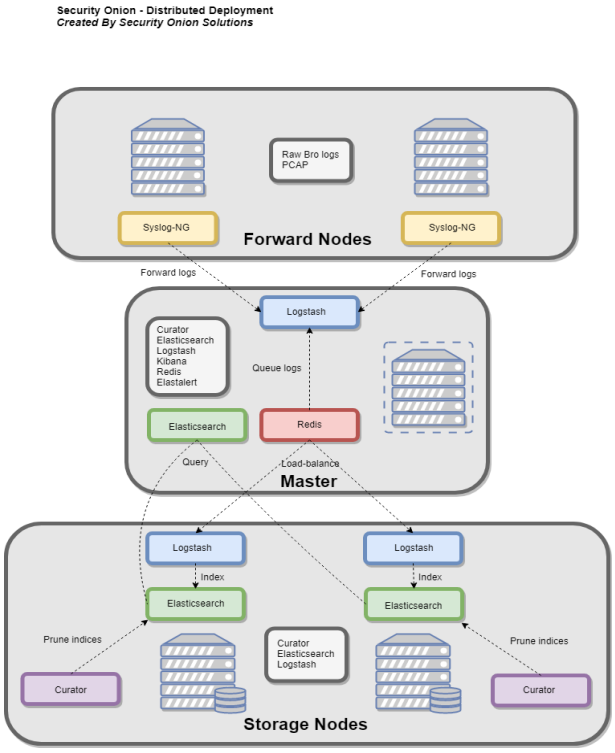
Security Onion on Ubuntu-pohjainen Linux-jakelu, joka sisältää kattavan määrän erilaisia työkaluja verkon valvontaan. Security Onion ei rajoitu olemaan vain tunkeutumisen havaitsemisjärjestelmä, vaan siitä käytetään nimitystä Network Security Monitoring platform (NSM), eli verkonvalvonta-alusta. (Morrow. 2019.) Se tarjoaa mm. täyden pakettien kaappauksen, konekohtaisen ja verkkopohjaisen tunkeutumisen havaitsemisjärjestelmän sekä useita sovelluksia kerätyn tiedon analysointiin. (Security Onion Solutions_a n.d.)

Riippuen käytössä olevasta laitteistosta sekä verkon koosta Security Onion ja sen työkalut voidaankin asentaa joko keskitetysti yhdelle (kuvio 1) tai hajautetusti useammalle palvelimelle (kuvio 2).



KUVIO 1. Security Onionin keskitetty asennus

Hajautetussa asennuksessa liikenteen havainnointi ja kaappaus, datan tallennus sekä datan indeksointi ja analysointi suoritetaan eri palvelimilla. Riippumatta siitä onko asennus tehty keskitetysti vai hajautetusti, ylläpitäjä voi visualisoida ja analysoida tietoja omalla koneellaan etänä. (Security Onion Solutions_c n.d.)



KUVIO 2. Security Onionin hajautettu asennus

4.1 Pakettien kaappaus

Pakettien kaappaukseen Security Onion käyttää netsniff-ng:tä. Netsniff-ng on avoimen lähdekoodin työkalu, joka perustuu Linuxin packet mmap -mekanismiin. Se voi tallentaa pcap-tiedostoja, toistaa niitä ja sen avulla voi tehdä niistä analyysejä. Se tukee myös 802.11 -kehysten kaappaamista, analysointia tai toistoa. Sillä luodut pcap-tiedostot ovat myös yhteensopivia tcpdumpin ja Wiresharkin kanssa. Netsniff-ng prosessoi pcap-tallenteet käyttäen joko scatter-gather I/O:a tai mmap I/O:a. (Netsnif-ng n.d.)

Netsniff-ng kaappaa kaiken verkkoliikenteen, joka Security Onion -sensoreille ohjataan ja tallentaa siitä tietoa niin paljon kuin tallennustilan puitteissa on mahdol-

lista. Security Onionissa on sisäänrakennettu mekanismi vanhan tiedon poistamiseksi tallennustilan täyttyessä. Security Onion kuvaa täyttä paketin kaappausta kuin valvontakameran kuvaksi, mutta sen avulla saadaan paljon enemmän tietoa hyökkääjästä ja hyökkäyksestä. Kaapattua dataa voidaan käyttää analysointiin Security Onionin IDS-järjestelmillä. (Security Onion Solutions_a n.d.)

4.2 IDS-järjestelmät Security Onionissa

Security Onion sisältää useita erilaisia IDS-vaihtoehtoja, joita voi käyttää sekä rinnakkain että erikseen. Sääntöpohjaisen NIDS:n osalta vaihtoehtoina ovat Snort ja Suricata, analyysipohjaisen NIDS:n osalta käytettävissä on Zeek ja HIDS:n osalta Wazuh. (Security Onion Solutions_a n.d.)

4.2.1 Snort

Snort on Roeschin kehittämä, nykyisin Sourcefiren ylläpitämä, avoimen lähdekoodin verkkopohjainen tunkeutumisen havaitsemisjärjestelmä. Se on alun perin luotu vuonna 1998 ja Roesch perusti sen kehittämistä varten vuonna 2001 Sourcefiren. (Carr 2007.) Vuonna 2013 Sourcefire siirtyi Ciscon omistukseen (Rao 2013).

Snort on GNU General Public License lisenssin alainen ohjelmisto, eli sen käyttö on ilmaista ja käyttäjät voivat muokata sitä, integroida sen omiin tuotteisiinsa ja levittää sitä GPL:n ehtojen mukaisesti. (Roesch 1999.) (Cisco_a n.d.)

Snort tarjoaa reaaliaikaisen verkkoliikenteen analysoinnin ja pakettien lokituksen IP-verkoissa. Se voi suorittaa protokolla-analyyseja, hakea ja yhdistellä sisältötietoja ja havaitsee useita erilaisia hyökkäystyyppejä kuten puskurin ylivuotoja ja porttiskannauksia. (Cisco_b n.d.)

Snort on sääntöpohjainen NIDS, joka käyttää sääntöjen ja esikääntäjien yhdistelmää liikenteen analysoimiseen. Säännöt tarjoavat yksinkertaisen ja joustavan tavan allekirjoitusten luomiseen yksittäisen paketin tutkimiseksi. Esikääntäjäkoodi

mahdollistaa laajemman tutkimuksen ja tietojen käsittelyn, jota ei voida tehdä pelkästään sääntöjen avulla. Esikääntäjät voivat suorittaa useita tehtäviä, kuten IP:n eheytys, porttiskannausten tunnistus, verkkoliikenteen normalisointi ja TCP-virran uudelleenkokoaminen. Esikääntäjät antavat Snortille kyvyn tarkastella ja käsitellä liikennettä laajemmin, kun säännöt käsittelevät yhden paketin kerrallaan. (Northcutt & Novak 2002.)

Käyttäjät voivat itse kirjoittaa sääntöjä, sekä ladata valmiita sääntöpaketteja Snortin palvelimilta. Näitä paketteja päivitetään jatkuvasti yhteisön ylläpitämänä. Parhaimmillaan uuden hyökkäyksen tullessa ilmi, sille luodaan sääntö samana päivänä. (Roesch 1999.) Sääntöpaketti sisältyy Snortin asennukseen ja rekisteröitymällä saa käyttöön sääntöjen päivityksen. Käyttäjä voi valita joko ilmaisen rekisteröitymisen tai maksullisen tilauksen. Maksullista vaihtoehtoa tarjotaan yksityisille käyttäjille sekä yrityksille eri hinnoin. Ilmaisversio päivitetään 30 päivän viiveellä maksullisesta. (Cisco_c n.d.)

Security Onionissa Snortin kanssa voi käyttää useita eri vaihtoehtoisia sääntöpaketteja tai näiden yhdistelmiä. Snortille ensisijaisesti optimoituja ovat Snortin ilmaiset Community- ja Registered-paketit sekä Snortin maksullinen Subscriber eli Talos-sääntökokoelma. Lisäksi käytettävissä ovat Suricatalle optimoidut Emerging Threadsin ilmaiset Open- sekä maksulliset Pro-säännöt. (Security Onion Solutions_d n.d.)

4.2.2 Suricata

Suricata on toinen Security Onionissa valittavissa oleva sääntöperustainen tunkeutumisen havaitsemisjärjestelmä. Snortin tavoin se on avoimen lähdekoodin ohjelmisto. Sitä kehittää Open Information Security Foundation (OISF) ja se on julkaistu 2010. (Sanders & Smith 2013.)

Uudempana IDS-moottorina sen tarkoitus ei ole ollut vain korvata olemassa olevia työkaluja, vaan tuoda uusia ideoita ja teknologioita esiin. Suricata mahdollistaa monisäikeistyksen, jonka avulla verkon liikenteen analysointi on tehokkaampaa ja nopeampaa. (Open Information Security Foundation_a n.d.)

Suricatan lähdekoodi on GNU GPLv2:n alainen ohjelmisto, kuten Snort, eli myös sen käyttö on ilmaista ja käyttäjät voivat muokata ja levittää sitä GPL:n ehtojen mukaisesti. (Open Information Security Foundation_b n.d.) (Open Information Security Foundation_c n.d.)

Vastaavasti kuin Snortille, on Security Onionissa myös Suricataan mahdollista valita eri sääntökokoelmia tai niiden yhdistelmiä. Emerging Threatsin Open- ja Pro-kokoelmat on optimoitu Suricatalle, sekä niiden ohella voi käyttää Snortin sääntöjä. Snortin Shared Object-säännöt toimivat kuitenkin vain Snortissa. (Security Onion Solutions_d n.d.)

4.2.3 Zeek

Security Onion käyttää verkon liikenteen analysointiin perustuvana IDS:nä Zeekiä. Zeek on aiemmin tunnetttu nimellä Bro, ja siihen viitataankin monissa järjestelmän osissa vielä vanhalla nimellään. (Security Onion Solutions_e n.d.)

Zeekin on alun perin kehittänyt Vern Paxson ja nykyisin sitä kehitetään International Computer Science Instituten projektina. (Zeek n.d.)

Zeek valvoo verkon liikennettä ja tallentaa lokitiedot kaikista mahdollisista yhteyksistä, DNS-kyselyistä, havaituista verkkopalveluista ja sovelluksista, SSL sertifikaateista, HTTP-, FTP-, IRC-, SMTP-, SSH-, SSL-, ja Syslog-tapahtumista. Lisäksi Zeek sisältää analysaattorit useille eri protokollille ja oletuksena mahdollisuuden verrata HHTP-latauksin MD5-tiivisteitä Team Cymrun Malware Hash Registry -projektin haittaohjelmiksi tunnettuihin tiivisteisiin.Zeekin havaitsemat tiedostot voidaan ohjata karanteeniin tai virustarkastukseen. (Security Onion Solutions_a n.d.)

Zeekin logit kerätään syslog-ng:llä, niitä jäsennellään ja täydennetään Logstashilla, ne tallennetaan Elasticsearchilla ja ne ovat nähtävissä Kibanalla (Security Onion Solutions_e n.d).

4.2.4 Wazuh

Versiosta riippuen Security Onionissa käytetään isäntäpohjaisena tunkelijan havaitsemisjärjestelmänä joko OSSECia tai Wazuhia (Security Onion Solutions_f, n.d). Versiosta 16.04 eteenpäin Wazuh on ollut Security Onionin pääasiallinen HIDS (Burks D 2018).

Wazuh on Wazuh Inc:n kehittämä ilmainen avoimen lähdekoodin HIDS, joka on saatavilla Windowsille, Linuxille ja macOS:lle. (Wazuh Inc_a n.d) Wazuh on OSSECin forkkaus eli haara. Sen kehittäjät katsoivat, ettei OSSEC ole saanut uusia ominaisuuksia, eikä sitä ole kehitetty tarpeeksi, vaan se on ollut lähinnä ylläpidetty. Siksi vuonna 2015 Wazuhin kehittäjät jatkoivat sitä omana projektinaan. (Wazuh Inc_b n.d.)

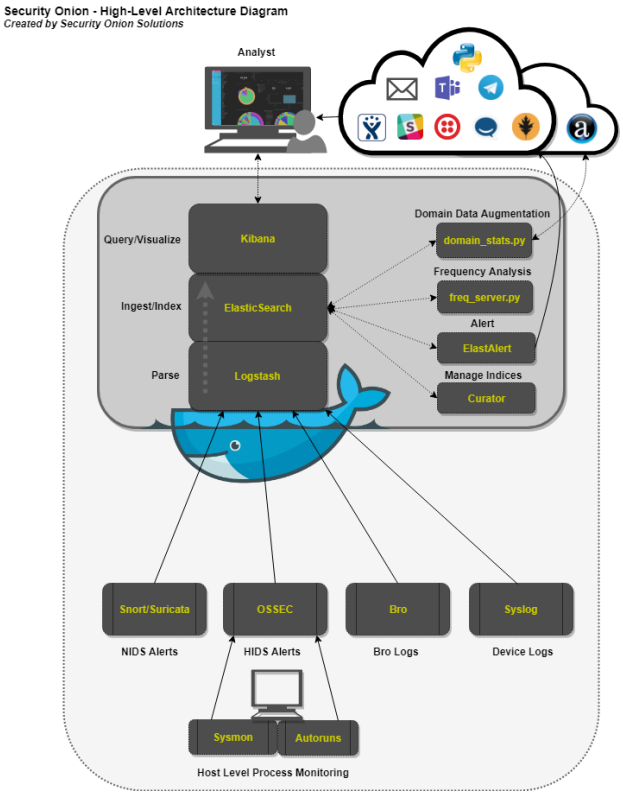
Security Onionissa sen tarkoitus on järjestelmän tarkkailu, eli suojata Security Onionia itseään tarkkailemalla sen eheyttä, tunnistamalla uhkia ja raportoimalla niistä. Sen lisäksi Wazuh-agentteja voi lisätä muille laitteille. Näitä voi lisätä maksimissaan 14000. (Security Onion Solutions_f n.d.)

4.3 Elastic Stack

Elastic Stack on Elasticin ylläpitämä avoimen koodin lokien hallintaympäristö. Elastic Stack on maailman suosituin lokien hallinta-alusta. Se sisältää työkalut lokien keräämiseen, indeksointiin, hakuun ja visualisointiin. Elastic Stack sai alkunsa Shay Banonin kehittämästä tietokantatyökalusta Elasticsearchista. Myöhemmin siihen lisättiin visualisointia varten Kibana ja lokitiedon keräämistä varten Logstash. Näiden sovellusten alkukirjaimista muodostui alustan alkuperäinen nimi ELK Stack. Kun siihen lisättiin Beats-sovellus, nimi vaihdettiin nykyiseen Elastic Stackiin. (Berman 2019.) (Gupta & Gupta 2017.)

Security Onionissa Elastic Stackista käytetään Logstashia ja Elasticsearchia IDS:ltä kerätyn datan jäsentelyyn ja indeksointiin sekä Kibanaa sen visualisoin-

tiin. Lisäksi apusovelluksina käytetään mm. Curatoria indeksien ajastettuun ylläpitoon. Kuviossa 3 kuvataan Elastic Stackin rakennetta Security Onionissa (Security Onion Solutions_c n.d.)



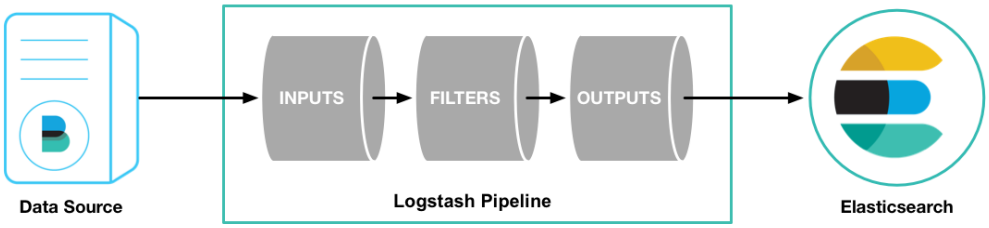
KUVIO 3. Elastic Stack Security Onionissa

4.3.1 Logstash

Logstash on avoimen lähdekoodin tiedonkeruumoottori. Se kykenee yhdistämään erillisistä lähteistä tulevaa tietoa dynaamisesti ja normalisoimaan kerätyn tiedon käyttäjän valitsemiin kohteisiin. Logstash sisältää useita plugineita tiedon lukemisen ja lähettämisen. Jos tietyn lähteen tietojen lukemista, datan kirjoittamista sijaintiin tai muokkaamista varten ei ole saatavilla sopivaa pluginia, käyttäjä voi kirjoittaa sille myös omiaan.

Logstash ei ainoastaan lähetä tietoja lähteestä kohteeseen, se myös muokkaa ja suodattaa kerättyä raakatietoa sekä muuntaa ne merkitykselliseksi, muotoilluksi ja järjestetyksi. Päivitetyt tiedot lähetetään sitten esim. Elasticsearchille. (Elastic_a n.d.) (Gupta & Gupta 2017.)

Logstash käyttää datan jäsentelyyn ns. putkitusta (kuvio 4), joka koostuu kahdesta vaaditusta elementistä, eli sisääntulosta ja ulostulosta, sekä valinnaisesta elementistä, suodattimesta (Elastic_d n.d.).



KUVIO 4. Logstashin putkitus

4.3.2 Elastisearch

Elasticsearch on avoimen lähdekoodin Apache Luceneen perustuva hakumoottori. Se tarjoaa reaaliaikaisen haun ja analysoinnin kaikille datatyypeille. (Elastic_b n.d.) Logstashin keräämä data lähetetään Elasticseachille indeksoitavaksi. Elasticsearchia ei kuitenkaan käytetä ainoastaan tiedon indeksointiin vaan se on myös täysi tekstihakumoottori, erittäin skaalautuva ja hajautettu järjestelmä. Se pystyy käsittelemään mm. tekstiä, numeerista dataa tai sijaintietoja. Tiedonhaun lisäksi informaatiota voidaan koota trendien ja kaavojen havaitsemiseksi. (Gupta & Gupta 2017.)

4.3.3 Kibana

Kibana on Elasticsearhin kanssa toimimaan suunniteltu analytiikka- ja visualisointinalusta. Muiden Elastic Stackin työkalujen tavoin se on myös avointa lähdekoodia. Sitä voidaan käyttää Elasticsearchin indekseihin tallennetun tiedon hakemiseen, tarkasteluun ja käsittelyyn. (Elastic_c n.d.) Kibana on selainpohjainen työkalu, joka käyttää Elasticsearhin rajapintoja, ja sillä voidaan lukea ja etsiä tietoa sen indekseistä. Sen avulla dataa voidaan visualisoida ja analysoida erilaisina kaavioin, graafein ja taulukoin. (Gupta & Gupta 2017.)

4.4 Security Onionin analysointityökalut

Tässä kappaleessa käydään läpi Security Onionissa käytettävissä olevia analysointityökaluja. Niitä voidaan käyttää yksittäin tai siirtyä työkalusta toiseen. Osaa käytetään suoraan Security Onionin käyttöliittymästä, osa on käytettävissä selaimella.

4.4.1 Sguil

Sguil on Bamm Visscherin kehittämä avoimen lähdekoodin analysointityökalu (Security Onion Solutions_a n.d). Sen keskeisin osa on graafinen käyttöliittymä, joka mahdollistaa pääsyn reaaliaikaisen tapahtumien seurantaan, istuntotietoihin ja raakatietojen pakettikaappaukseen. Se helpottaa verkkoturvallisuuden valvontaa ja tapahtumapohjaista analyysia. (Visscher 2014.)

Security Onionissa Sguil tarjoaa yhden selkeän käyttöliittymän Snortin, Suricatan ja Wazuhin tuottamien hälytysten tarkasteluun (Security Onion Solutions_f n.d). Sguilin kautta kerättyä dataa voi tarkistella sen sisäänrakennetulla trancscript-työkalulla, joka esittää pakettikaappauksen tiedot ASCII-muodossa tai kääntämällä sen muihin Security Onionin työkaluihin, kuten Wiresharkiin, NetworkMineriin tai Kibanaan. (Security Onion Solutions_g n.d.)

4.4.2 Squert

Squert on Paul Hallidayn alunperin kehittämä selainpohjainen käyttöliittymä Sguilin tietokantojen käyttöön (Security Onion Solutions_i n.d.). Security Onion ylläpitää omaa versiotaan Squertista, koska Halliday on luopunut sen kehittämisestä (Security Onion blog 2016).

Squert ei kuitenkaan ole reaaliaikainen. Squertista on mahdollista siirtyä täyteen paketinkaappaukseen käyttäen CapMe:ta. (Security Onion Solutions_a n.d.)

4.4.3 Kibana

Security Onionissa Kibanaa voidaan käyttää muiden työkalujen keräämään datan analysointiin ja visualisointiin. Kibanan kautta voidaan tarkkailla verkon liikennettä ja tutkia paitsi IDS-lokeja, myös Zeekin lokeja sekä järjestelmälokeja. Myös Kibanasta on mahdollista siirtyä täyteen paketinkaappaukseen käyttäen CapMe:ta. (Security Onion Solutions_a n.d.).

4.4.4 CapMe

CapMe on Paul Hallidayn kehittämä työkalu paketinkaappaustietojen tarkasteluun (Security Onion Solutions_a n.d.).
CapMe on web-pohjainen käyttöliittymä, jonka avulla voi tarkastella tcpflow:n tai Zeekin luomia paketinkaappaustietoja sekä ladata täysiä PCAP-tiedostoja. CapMehen voi kääntää dataa Squertin NIDS-hälytyksistä tai Kibanan logeista joissa on aikaleima, lähteen IP-osoite ja portti sekä kohteen IP-osoite ja portti.
(Security Onion Solutions_h n.d.)

4.4.5 CyberChef

CyberChef on Yhdistyneen kuningaskunnan tiedustelu- ja turvallisuuspalvelun Government Communications Headquartersin, GCHQ:n kehittämä selainpohjainen avoimen lähdekoodin sovellus datan analysointiin ja dekoodaukseen. Sen kehittäjät kutsuvat sitä sveitsiläiseksi linkkuveitseksi ja se on pyritty kehittämään helppokäyttöiseksi ja intuitiiviseksi, mutta myös ammattikäyttöön soveltuvaksi. (GCHQ 2016.)

CyberChefllä voi muuntaa erilaisia datatyyppejä toisiin. Se tukee niin yksinkertaisempia salausmenetelmiä, kuten XOR:ia ja Base64:ää, kuin modernimpia tehokkaampia salausalgoritmeja, kuten AES:ia, DES:ia ja Blowfishia. CyberChefia voidaan käyttää monissa tilanteissa, kuten Base64-koodattujen merkkijonojen dekoodamiseen, päivämäärien ja ajan muuntamiseen toiseen aikavyöhykkeeseen, IPv6-osoitteiden jäsentämiseen, hexdumpien purkamiseen tekstiksi ja salauksen purkamiseen. (CyberChef Readme n.d,)

Tämän lisäksi se voi suorittaa valtavan määrän muita erilaisia operaatioita. CyberChef ei ole kuitenkaan pelkkä datatyypin muunnin, vaan sen lisäksi sillä voi ketjuttaa operaatioita niin, että edellinen operaatio toimii lähteenä seuraavalle. Nämä operaatiot voi tallentaan ”resepteinä”, joita voi käyttää myöhemmin.

4.4.6 Wireshark

Pakettien sisällön analysointiin Security Onion tarjoaa Wiresharkin, joka on maailman yleisimmin käytetty pakettianalysaattori. Sen on alun perin luonut Gerald Combs vuonna 1998 (Security Onion Solutions_j, n.d). Wireshark tunnettiin alun perin nimellä Ethereal, mutta nimi vaihdettiin vuonna 2006 tekijänoikeussyitä Wiresharkiin. Wireshark on GNU lisenssin alainen avoimen lähdekoodin ohjelmisto, ja sen käyttö myös liiketoiminnassa on ilmaista. (Wireshark FAQ n.d.)

Wireshark on erittäin monipuolinen työkalu ja siinä on lukematon määrä erilaisia ominaisuuksia. Se tukee satoja eri protokollia ja tuettuja protokollia lisätään jatkuvasti. Sitä voi käyttää sekä datan reaaliaikaiseen kaappaukseen että offline-analysointiin.

Wireshark tukee monipuolisesti eri lähteitä ja datatyyppejä. Riippuen ympäristöstä Wireshark kykenee lukemaan reaaliaikaisesti dataa mm. ethernetistä, wlan-verkoista, bluetoothista, USB:stä sekä nykyisin vähemmän käytetyistä verkoista, kuten Token Ringistä. Wireshark pystyy myös lukemaan ja kirjoittamaan useita erilaisia tiedostotyyppejä, kuten libpcap ja Pcap NG. Analysointia helpottaa suodattimet, joilla voi rajata näytettäviä tuloksia. Myös VoIP-liikenteen analysointi on mahdollista. Helppokäyttöisyyden vuoksi Wiresharkissa on myös tuki TTY-tilalle kuulovammaisia varten graafisen käyttöliittymän ohella. (Wireshark n.d.)

Wiresharkissa on kuitenkin havaittu useita tietoturva-aukkoja, joten sen päivitysten ajantasaisuudesta täytyy huolehtia ja sen käyttöä epäturvallisissa verkoissa välttää (SecTools n.d).

Security Onionissa Wiresharkin voi avata suoraan sovellusvalikosta tai siihen voi siirtyä Sguilista (Security Onion Solutions_j n.d).

4.4.7 NetworkMiner

Kaapattujen pakettien lisäanalysointia varten Security Onionissa voidaan käyttää NetworkMineria. NetworkMiner on avoimen lähdekoodin Network Forensic Analysis Tooliksi (NFAT) kutsuttava sovellus. Se on tarkoitettu käytettäväksi Windowsilla, mutta on saatavilla myös muille käyttöjärjestelmille. Sitä voi käyttää passiivisena pakettien kaappaustyökaluna, jolla voi havaita käyttöjärjestelmiä, sessiotietoja, isäntänimiä, avoimia portteja ja muita tietoja luomatta liikennettä verkkoon. Sen lisäksi sillä voi avata pcap-tiedostoja ja poimia niistä siirrettyjä tiedostoja. NetworkMiner on saatavilla maksullisena sekä toiminallisuudeltaan rajoitetumpana ilmaisversiona. (Netresec n.d.)

NetworkMineria on mahdollista käyttää reaaliaikaiseen paketinkaappaukseen, mutta se on soveltuvampi offline-analysointiin (Hjelmvik, n.d). Wiresharkista poiketen NetworkMiner keskittyy pakettien raakatietojen sijaan laitteisiin, kuten palvelimiin ja niiden ominaisuuksiin (SecTools n.d).

Se poimii pcap-tiedostosta IP-osoitteiden mukaan listattuna tiedot mm. laitteen MAC-osoitteesta, käyttöjärjestelmästä, DNS-nimestä ja käytetystä portista. Osa näistä tiedoista on helppo saada suoraan kaapatusta paketista, osa joudutaan päättelemään vertaamalla tietoja muiden sovellusten tietokantoihin. Näistä tiedoista voidaan hakea hyökkäykseen käytettyjä tai hyökkäyksen kohteeksi joutuneita koneita. (Hjelmvik n.d.)

NetworkMinerilla myös mahdollista poimia ja koota tiedostoja kaapatusta verkkoliikenteestä. Tuettuja protokollia tälle ovat mm. http, SMB ja FTP. Tällä voidaan selvittää, mitä tiedostoja hyökkääjä mahdollisesti on onnistunut lataamaan koneelta tai onko koneelle ladattu haittaohjelmia. Lisäksi NetworkMinerin avulla on mahdollista selvittää vuotaako koneelta tietoja verkkoon. (Hjelmvik n.d.)

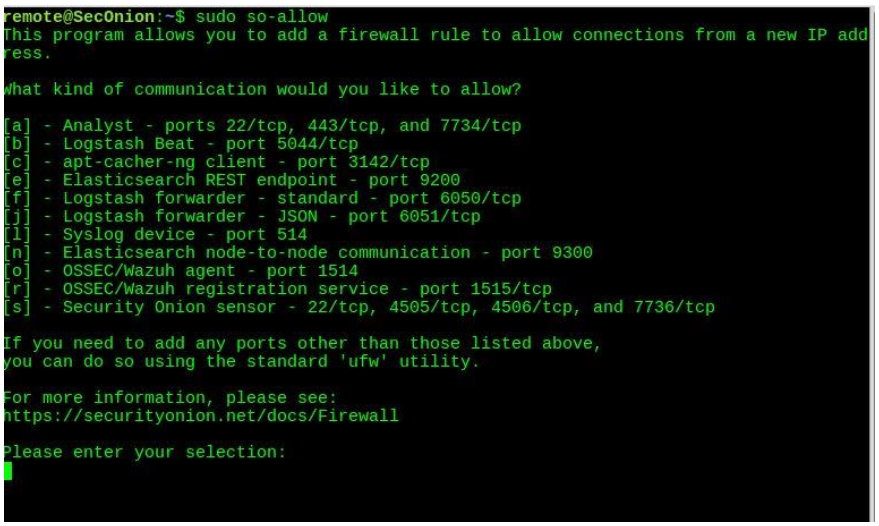
Security Onionissa NetworkMinerin voi avata suoraan sovellusvalikosta tai siihen voi siirtyä Sguilista (Security Onion Solutions_k n.d).

5 ANALYSOINTI

Security Onion mahdollistaa sen keräämän datan analysoinnin usein eri tavoin. Yksinkertaisin keino on suoraan koneelta tai palvelimelta, jolle Security on asennettu. Tämä tarjoaa pääsyn kaikkiin analysointityökaluihin, kuten Sguiliin. Security Onion ei kuitenkaan tätä suosittele (Security Onion Solutions_l n.d).
Tämän työn käytännön toteutuksessa etäyhteys Security Onion -palvelimelle sallittiinkin pääasiassa hallintaa varten, ei analysointia. Security Onion suositteleekin analysointiin ylläpitäjän omalle työasemalle luotua virtuaalikonetta, jolle Security Onion sekä sen työkalut asennetaan analysointia varten (Security Onion Solutions_m n.d).
Tämän lisäksi voidaan käyttää selainpohjaisia työkaluja, kuten Kibanaa tai Squertia, joihin voi ottaa yhteyden millä tahansa selaimella, mistä tahansa samassa verkossa olevalta laitteella.

5.1.1 Analysointi virtuaalikoneen kautta

Security Onion suosittelee kerätyn tiedon analysointiin ylläpitäjän omalle työasemalle erikseen luotua virtuaalikonetta. Se asennetaan samoin kuin Security Onion normaalistikin, mutta asennuksen jälkeen jätetään verkkoliitäntöjen ja IDS:n määrittäminen suorittamatta. Näin käytettävissä on Wiresharkin, NetworkMinerin ja Sguilin paikallisesti asennetut versiot. Niillä voidaan ottaa yhteys Security Onion-palvelimeen, jolta pcap-tiedostot ladataan analysoitavaksi.
Yhteyden sallimiseksi täytyy avata sitä varten vaadittavat portit UFW-säännöissä so-allow-komennolla. Komento avaa valikon (kuvio 5), josta valitaan *analyst*-optio, joka avaa portit 22, 443 ja 7734 TCP-liikenteelle. Tämän jälkeen valitaan joko yksittäinen IP-osoite tai aliverkko, josta liikenne halutaan sallia. Valikosta voi valita myös muita portteja eri toimintoja varten.



KUVIO 5. Allow-valikko

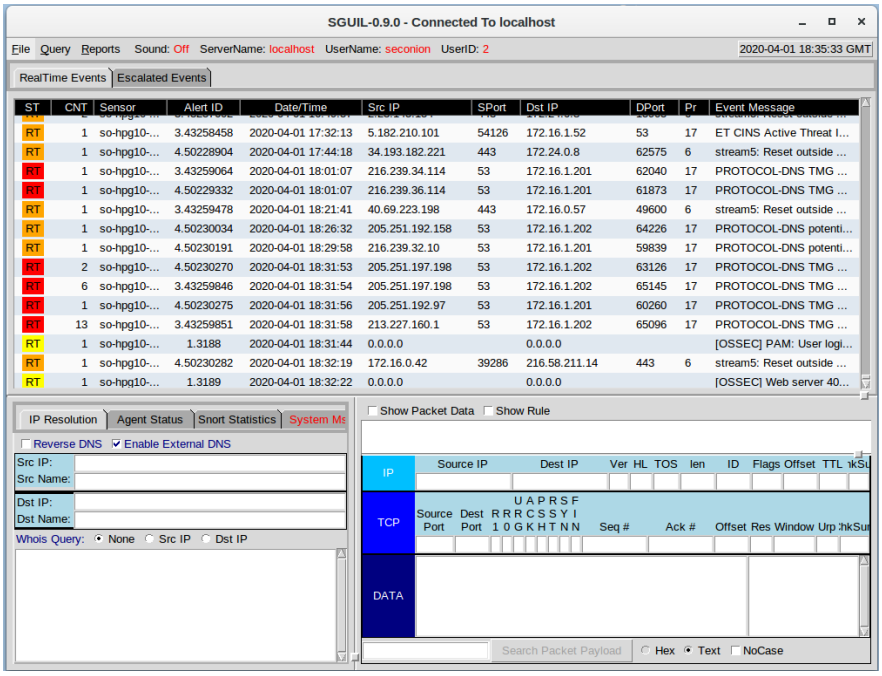
Kun liikenne on sallittu, voi luodulta virtuaalikoneelta avata tarvittavan sovelluksen, kuten Sguilin jolta voi ottaa yhteyden Security Onion-palvelimen IP-osoitteeseen tai isäntänimeen. Sguil näyttää palvelimen keräämän tiedon aivan kuten suoraan palvelimelta suorittaessa. Samoin voidaan käyttää selainpohjaisia työkaluja kuten Kibanaa tai Squertia. Käytetystä työkalusta voi edelleen avata pcap-tiedostoja tarkasteltavaksi esim. Wiresharkissa

5.2 Analysointi Sguilissa

Sguil on Security Onionin keskeisin työkalu, johon kerätään kaikki Snortin, Suricatan, Zeekin ja Wazuhin tuottamat hälytykset. Niitä voi tarkastella suoraan Sguilissa Zeekin luomaa transkriptia käyttäen, sekä edelleen kääntää Wiresharkiin tai NetworkMineriin. Sguilista voi myös tehdä hakuja suoraan erilaisista lähteistä käyttäen IP-osoitetta, porttia tai hälytyksen aiheuttanutta sääntöä. Lisäksi tapahtumia voi korottaa tarkempaa analysointia varten.

Sguilin avatessa käyttäjä voi valita minkä sensoreiden hälytykset näytetään. Perusnäkymässä Sguil näyttää valinnan mukaan Snortin tai Suricatan NIDS-hälytykset riippuen siitä kumpi on käytössä sekä Wazuhin HIDS-hälytykset. Sguil ryhmittelee hälytykset tyypin mukaan ja tulokset on mahdollista järjestää mm. tilan, lukumäärän, sensorin, lähteen tai kohteen IP-osoitteen, portin tai tapahtumahetken perusteella. Mikäli samankaltaisia hälytyksiä on useampia, voi ne avata uuteen välilehteen klikkaamalla lukumääräsaraketta ja valitsemalla *View correlated*

events. Päänäkymässä voi myös valita näytetäänkö hälytyksen paketin data ja hälytyksen aiheuttaneen säännön tiedot. Kuviossa 6 Sguilin perusnäkymä, jossa näkyy Security Onionin keräämiä hälytyksiä.



KUVIO 6. Sguil

Hälytyksen ID:tä hiiren kakkospainikkeella napauttamalla voi avata kaappauksen transkriptin tai avata pcap-tiedoston Wiresharkissa tai NetworkMinerissa. Transkriptin voi avata myös keskimmäisellä näppäimellä. IP-osoitetta tai porttia klikkaamalla saa auki valikon, josta voi valita erilaisia hakuja tai kopioida osoitteen. Hakuja voi suorittaa niin Sguilin tuloksista, kuin ulkoisista tietokannoista tai hakukoneista.

5.2.1 Hakutoiminnot

Sguilin sisällä voi tehdä pikahaun lähde- tai kohde-IP-osoitteen tai -portin perusteella. Tällöin Sguil hakee tapahtumat, joissa on sama osoite tai portti. Advanced query -haulla voi käyttää myös tarkempia hakuehtoja, ja AND-, OR-, ja NOT-operaattoreita.

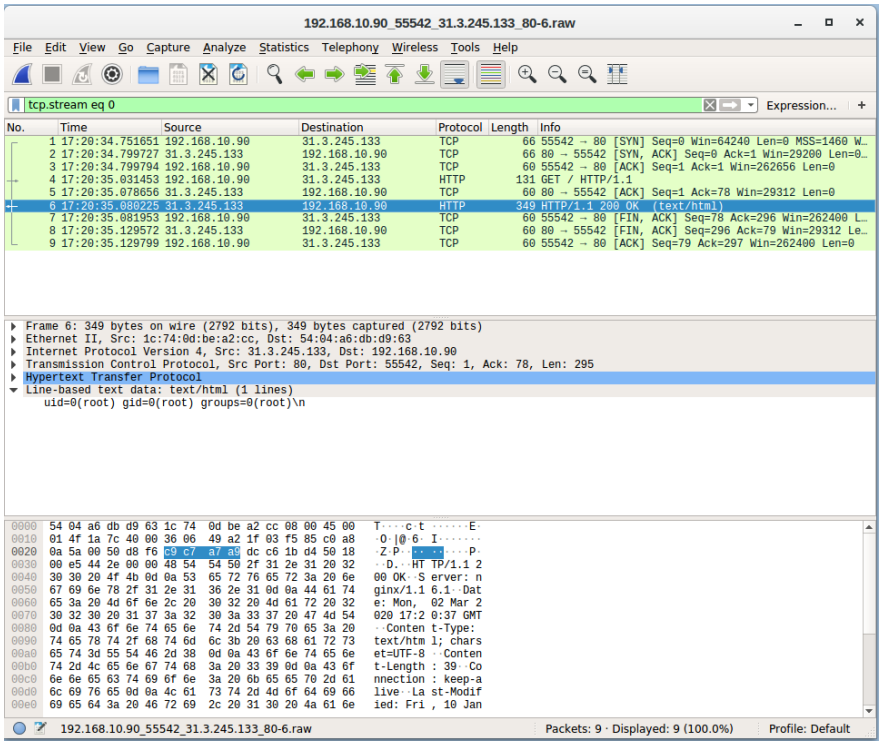
Ulkoisilla hauilla on mahdollista käyttää suoraan Sguilista erilaisten tietoturvasivustojen tietokantoja tai Googlen hakua. Sguil avaa Internet-selaimen ja yhdistää

valitulle verkkosivulle, ilman, että IP-osoitetta tarvitsee kopioida erikseen. Tietokantoja on käytettävissä useita, kuten Virustotal, Dshield, Alexa ja DomainTools. Hausta voi myös kääntää tiedot Kibanaan.

5.2.2 Pakettikaappauksen analysointi

Pcap-tiedoston avaaminen analysointia varten onnistuu suoraan Sguilista. Hälytyksen ID:tä klikkaamalla aukeavasta valikosta voi siirtyä Wiresharkiin tai NetworkMineriin.

Wireshark tarjoaa valtavan määrän keinoja analysoida kaapattua dataa. Päänäkymässään näyttää käytetyt kaapattujen kehysten aikaleimat, kohde- ja lähde-IP-osoitteet, protokollat, pituuden bitteinä, sekä tiedot liikenteestä. Kuviossa 7 Wiresharkin päänäkymä paketinkaappauksesta, jossa luotu hälytys terminaalissa curl-komennolla testmyids.com-sivustolta.

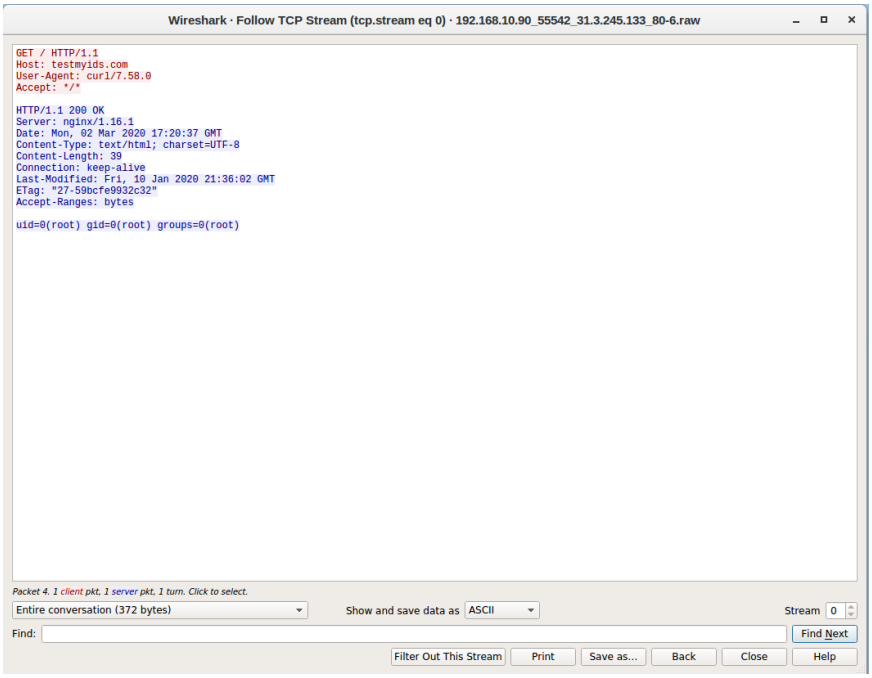


KUVIO 7. Wireshark

Statistics-valikosta voi valita erilaisia tilastoja kaapatusta liikenteestä. Protocol hierarchy näyttää tietoja käytetyistä protokollista. Endpoints näyttää liikenteen

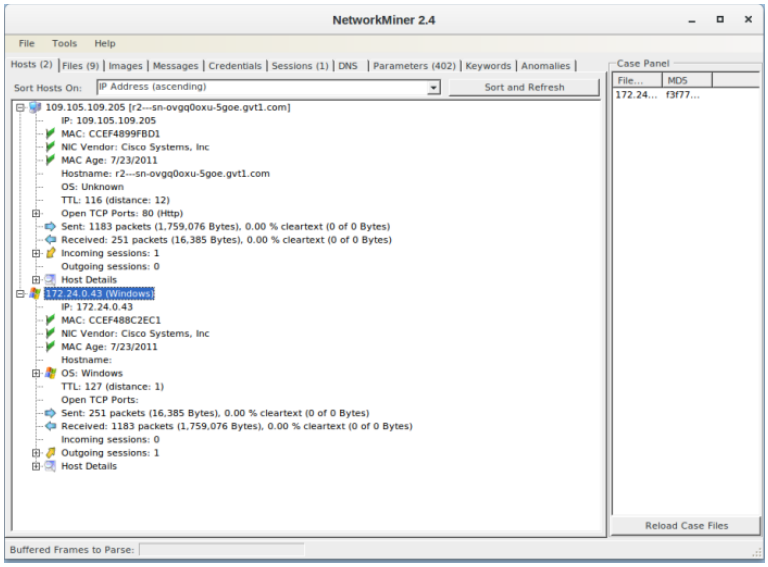
päätepisteiden mac- ja IP-osoitteet. Flow chartilla voi tarkastella liikenteen flow-tietoja.

Analyze-valikosta voi suodattaa tuloksia ja käyttää erilaisia analysointivaihtoehtoja. Follow TCP/HTTP Stream -toiminnolla Wireshark näyttää palvelimen ja koneen välisen liikenteen. Kuviossa 8 näkyy testmyids.com-sivun palauttaman vastauksen ” uid=0(root) gid=0(root) groups=0(root)”, joka on laukaissut Snortin hälytyksen.



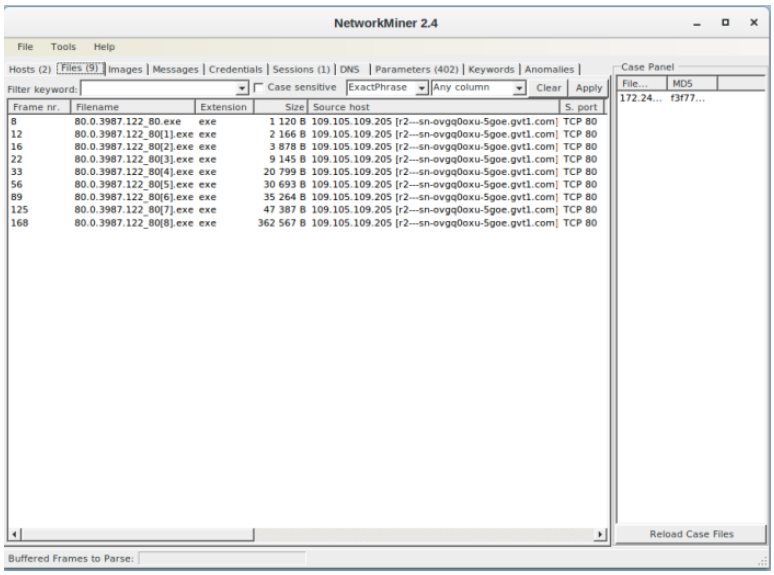
KUVIO 8. Wireshark - Follow TCP/HTTP Stream

NetworkMinerin Hosts-välilehdellä (kuvio 9) näytetään IP-osoitteet, verkkoadapterin MAC-osoite ja siitä tunnistettu valmistaja, valmistusvuosi, DNS-nimet, käyttöjärjestelmä sekä muita tietoja laitteista



KUVIO 9. NetworkMiner - Hosts

FIles-välilehdellä (kuvio 10) näytetään siirretyt tiedostot. Tiedostoa klikkaamalla voi avata valitun tiedoston, sen sijaintikansion tai laskea sen MD5-, SHA1- tai SHA256-tiivisteen. Avaamalla tallennuskansion, tiedoston voi ladata raahaamalla Virustotaliin ja verrata sen tietokantaan haittaohjelmien tai virusten varalta.



KUVIO 10. NetworkMiner - Files

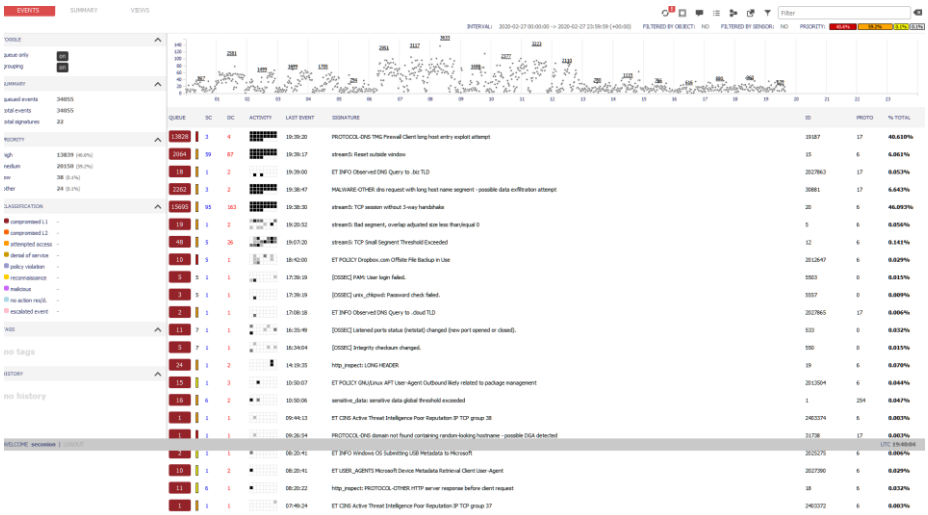
5.3 Selainpohjaiset analysointityökalut

Selainpohjaisiin työkaluihin voi ottaa yhteyden millä tahansa selaimella, mistä tahansa samassa verkossa olevalta laitteella, kun Security Onionista on avattu analysointia varten tarvittavat portit. Tärkeimmät selaimella käytettävät työkalut ovat Squert ja Kibana.

5.3.1 Analysointi käyttäen Squertia

Squertiin yhdistäminen onnistuu avaamalla selaimella osoitteen https://security-onion/squert, josta ”securityonion” korvataan joko Security Onionin IP-osoitteella tai DNS-nimellä.

Squertiin kirjaudutaan samoilla käyttäjätunnuksilla kuin Sguiliin. Päänäkymässä Squert näyttää käyttäjälle sekä NIDS:n että HIDS:n hälytykset Sguilia vastaavalla tavalla (kuvio 11).



KUVIO 11. Squert

Tietoja voi suodattaa ja järjestää monin eri tavoin. Oletuksena Squert näyttää aktiivisena jonossa olevat tapahtumat ryhmitettyinä tyypin mukaan. Queue only -asetusta muuttamalla näytetään kaikki hälytykset, ja grouping -asetuksen poistamalla jokainen hälytys näytetään erikseen. Käyttäjä voi valita myös näytetäänkö NIDS:n, HIDS:n vai molempien hälytykset. Tiedot voi valita näytettävän tietyltä aikaväliltä sekä mm. lähteen tai kohteen IP-osoitteen tai portin mukaan.

Hälytyksen Event ID:n valitsemalla pystyy kääntämään datan CapMe:en, josta voi tarkastella Zeekin luomaa transkriptia (kuvio 12) tai ladata koko pcap-tiedoston, esim. Wiresharkissa tarkasteltavaksi.

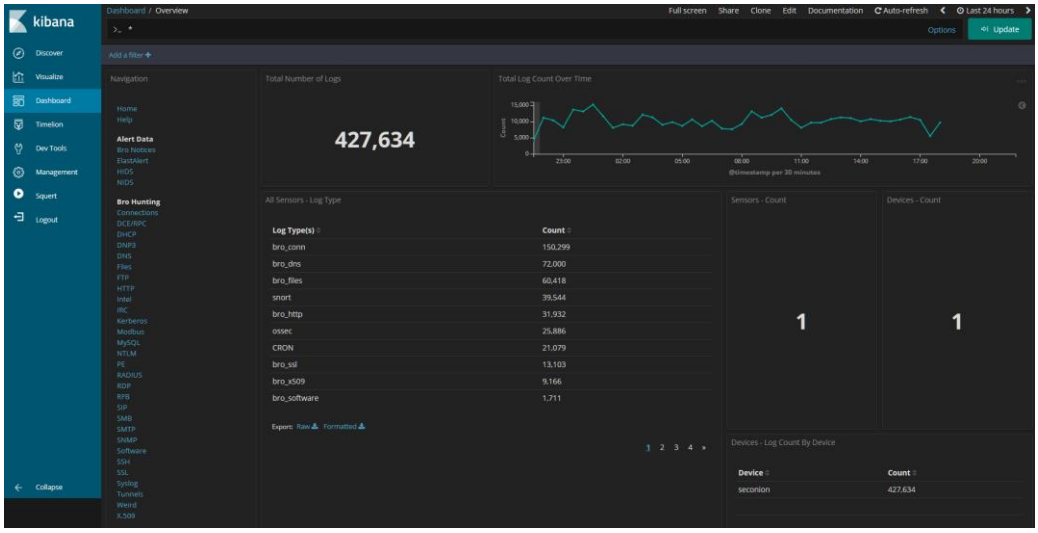
KUVIO 12. Zeekin transkripti CapMesta avattuna

IP-osoitetta, porttia tai signaturea klikkaamalla voi siirtyä tarkastelemaan tapahtuman tietoja Kibanassa tai etsiä siitä tietoja esim. Google-haulla tai Virustotalin sivuilta.

5.3.2 Visualisointi ja analysointi Kibanassa

Kuten Squertiinkin, Kibanaan yhdistäminen tapahtuu avaamalla selaimella osoitteen https://securityonion/app/kibana. Tämä avaa Kibanan Dashboard-näkymän. Kibanan sisällä on helppo siirtyä näkymästä toiseen. Vasemmassa laidassa on palkki, josta voi valita käytettävän toiminnon. Kibanan ensisijainen näkymä on Dashboard, jossa käytettävät kojelaudat ovat listattuna vasempaan laitaan. Security Onionissa Kibanaan on luotu useita valmiita kojelautoja. Kojelautoja voi myös itse muokata ja luoda lisää. Tarkemmin tietoja voi etsiä Discover-osiossa useita erilaisia hakuehtoja käyttäen. Siitä voi edelleen siirtyä visualisointiin. Lisäksi Kibanassa voi luoda itse visualisointeja tai aikajana-analyyseja tai siirtyä Squertiin.

Kibanan Overview-kojelaudalla (kuvio 13) näytetään tietoja valitulta aikaväliltä kerätyistä lokeista. Saatavilla on mm. listattuna lokien kokonaismäärä, niiden lukumäärät sensoreittain ja tyypeittäin sekä IDS-hälytysten määrät ja tyypit. Visualisoituna on graafi kerätyistä lokeista ajan mukaan sekä, Zeekin lokit datatyypeittäin ja ympyrädiagrammeina esitettynä maittain. Kojelaudoilla esitetyt lokitiedot on mahdollista tallentaa csv-tiedostoina.



KUVIO 13. Kibana - Overview

Eri tiedonsiirtoprotokollille ja IDS-järjestelmille luoduista kojelaudoista löytyy vastaavat visualisoinnit ja listaukset tarkemmin. Esimerkiksi NIDS-kojelaudalle on kerätty tiedot Snortin tai Suricatan hälytyksistä. IP-osoitetta klikkaamalla voi siirtyä edelleen kyseisen osoitteen lokitietoihin. Kibanasta voi myös siirtyä CapMehen klikkaamalla hälytyslokin *_id*-kenttää.

6 KÄYTÄNNÖN TOTEUTUS

Security Onionin asentaminen on nykyisin melko suoraviivainen prosessi, mutta kuten edellä on todettu, Security Onion sisältää valtavan määrän työkaluja. Siksi sen käyttöönottoon ei riitä, että se vain asennetaan ja annetaan olla. Ennen asennusta järjestelmää testattiin ja selvitettiin käyttöön soveltuva käytettävien työkalujen ja asetusten kokoonpano. Fyysisen verkon osalta piti saada liikenne ohjattua Security Onionille, niin ettei se aiheuta pullonkauloja tai muutoin häiritse verkon normaalia käyttöä. Lisäksi järjestelmää pitää olla mahdollista myös hallita ja käyttää etänä useammasta kohteesta.

6.1 Valmistelu

Ennen lopullista toteutusta Security Onionia testattiin virtuaalikoneilla. Koska WPK-verkon ensisijaisena virtualisointiympäristönä toimii Microsoftin Hyper-V, testaus toteutettiin sitä käyttäen. Tämä kuitenkin aiheutti tiettyjä ongelmia, johtuen siitä miten virtuaalisia verkkokortteja käsitellään Hyper-V:ssä. Niiden saaminen promiscous-tilaan ei ole suoraan mahdollista, vaan vaatii asetusten muokkausta Powershellissa. Security Onionia onkin tuotantototeutuksissa tästä johtuen suositeltavampaa käyttää VMWaren tai Oraclen VirtualBox-virtualisointiympäristöissä, joille Security Onion tarjoaa myös ohjeistuksen asennusta varten (Security Onion Solutions_k n.d).

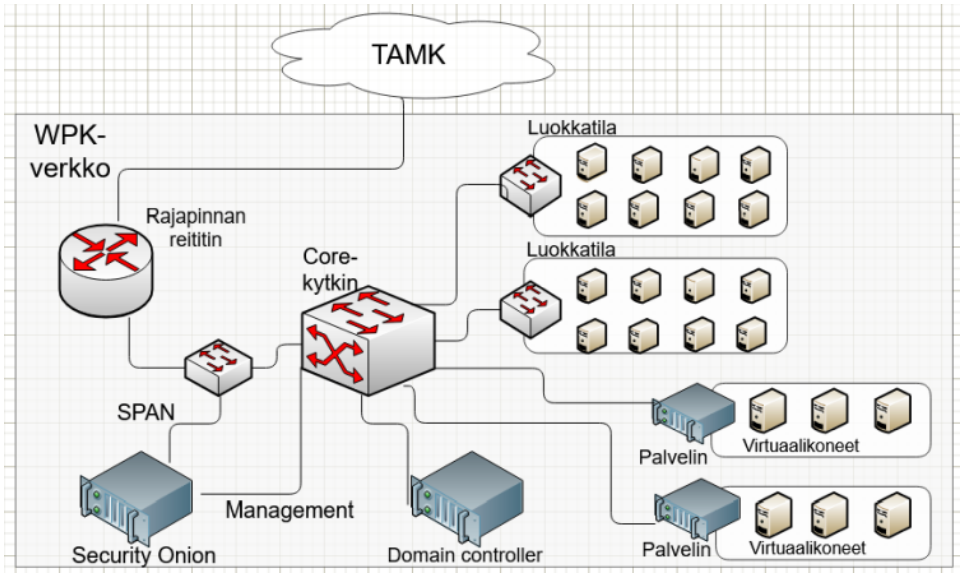
Testauksen tarkoituksena oli löytää sopivin mahdollinen kokoonpano toimeksiantajan tarpeisiin. Koska järjestelmä tulisi verkon valvonnan ohella myös opetuskäyttöön, sen pitäisi kyetä havaitsemaan mahdollisimman paljon potentiaalisia uhkia sekä tuottaa mahdollisimman paljon hälytyksiä näistä havainnoista.

Tätä tarkoitusta varten luotiin kaksi virtuaalikonetta, jotka käyttivät samoja fyysisiä verkkoliitäntöjä, sekä samoja virtuaalisia verkkokortteja. Toinen asetettiin käyttämään Snortia ja toinen Suricataa. Molempiin valittiin samat sääntökokoelmat, eli Snortin Community- ja Registered-paketit sekä Emerging Threadsin Open-kokoelma. Näihin päädyttiin niiden maksuttomuuden vuoksi, sekä mahdollisimman laajan sääntövalikoiman saavuttamiseksi. Näiden kahden välillä tehdyn

vertailun jälkeen päädyttiin valitsemaan lopulliseen asennukseen Snortia käyttävä kokoonpano. Se antoi testijakson aikana selkeästi suuremman määrän havaintoja ja hälytyksiä.

6.2 Asennus

Lopullinen Security Onionin asennus tehtiin omalle sitä varten varatulle palvelimelleen. Palvelimelta varattiin Security Onionille kaksi fyysistä verkkoliitäntää. Toinen on järjestelmän hallintaa varten käytettävä management interface ja toinen verkon valvontaa varten tarkoitettu sniffing interface. Security Onion suosittelee asettamaan hallintaa varten staattisen IP-osoitteen, mutta WPK-verkon osoiteasetusten vuoksi tämä ei ollut mahdollista, vaan sille varattiin DHCP-palvelimelta IP-osoite, ja Security Onionin asetuksissa IP-osoitteen määritykseen valittiin käytettävän DHCP:tä. Kuviossa 14 yksinkertaistettu topologiakuva verkon rakenteesta ja Security Onionin sijainnista.



KUVIO 14. WPK-verkko

Koska Security Onion on Ubuntu-pohjainen Linux-jakelu, itse järjestelmän asennus on varsin helppoa graafisen asennusohjelman avulla. Asennuksen voi tehdä käyttäen Security Onionin ISO-tiedostoa virtuaalilevynä tai luoda siitä fyysisen tallennusvälineen, jolta ohjelmistot asennetaan. Asennukseen voi käyttää myös haluamaansa Ubuntu-jakelua, jolle asennetaan Security Onionin osat.

Linux-jakeluna ja graafiselta käyttöliittymältään varsin riisuttuna Security Onionin käyttöjärjestelmä ei vaadi kovinkaan tehokasta konetta asennusta varten, mutta varsinkin täyden paketinkaappauksen ja IDS:n käyttö kuluttaa erittäin paljon resursseja. Koska Security Onion on asennettu suoraan fyysiseen ympäristöön, eikä virtuaalikoneelle, oli mahdollista varata kaikki saatavilla olevat resurssit sen käyttöön.

Asennusta tehdessä voi valita kuinka montaa rinnakkaista Snort-, Suricata-, tai Zeek-prosessia samanaikaisesti ajetaan, ja mitä enemmän niitä on, sitä enemmän järjestelmä vaatii muistia ja prosessointitehoa. Samalla kuitenkin saadaan enemmän ja luotettavammin tietoa liikenteestä, kun paketteja ei jouduta jättämään huomioimatta

Vaikka asennus itsessään on helppoa, haasteen kuitenkin asennukseen aiheutti tallennustila, koska asennukseen käytetyllä palvelimella oli useita fyysisiä levyjä, joista on luotu kaksi erillistä loogista levyä. Alkuperäinen tarkoitus oli käyttää toista järjestelmän asennukseen ja toista kerätyn datan tallennukseen. Tämä osoittautui kuitenkin hankalaksi, koska se olisi vaatinut tiedostosijaintien uudelleenmounttaamista. Security Onion ei mahdollista suoraa tietojen tallennuskohteen valintaa. Lopulta päädyttiin käyttämään Ubuntun Logical Volume Managementia, jolla levyt voidaan yhdistää yhdeksi osioksi ja käyttää sekä järjestelmän että tietojen tallennukseen.

6.3 Käyttöönotto

Järjestelmän asennuksen jälkeen tulee vielä suorittaa kaksivaiheinen asetusten määrittely. Ensimmäisessä vaiheessa määritellään verkkoliitännät ja toisessa käytettävien työkalujen, kuten IDS:n asetukset.

Verkkoliitäntöjen osalta management interface asetettiin hakemaan IP-osoitteensa verkon DHCP-palvelimelta ja sniffing interfaceksi asetettiin verkkokortti,

joka on kytketty kytkimeen, josta tarkkailtava liikenne ohjataan Security Onionille. Tälle liitännälle ei anneta IP-osoitetta.

Järjestelmän uudelleenkäynnistyksen jälkeen seuraavassa vaiheessa valitaan käytettävät työkalut ja määritetään niiden asetukset. Asennusohjelma antaa valita joko kokeiluversion eli Evaluation moden tai täyden version eli Production moden. Evaluation modessa Security Onion käyttää Security Onionin oletusasetuksia ja käyttäjän täytyy antaa vain käyttäjänimi ja salasana, joita käytetään Sguilissa, Kibanassa ja muissa työkaluissa.

Production Modessa voi myös valita Security Onion käyttävän oletusasetuksia Best Practices -valinnalla, mutta myös kustomoida asetukset itse. Oletusasetukset soveltuvat useimpiin käyttökohteisiin ja olisivat riittäneet tässäkin tapauksessa, mutta niihin haluttiin tehdä pieniä muokkauksia.

Sguilin tietokanta asetettiin säilytettävän oletusasetusten mukaisesti 30 vuorokautta, mikäli tallennustila ei täyty tätä ennen. DAYSTOREPAIR -asetuksella valitaan aika jolta Security Onion korjaa Sguilin tietokannan. Tätä varten järjestelmä suorittaa päivittäin automaattisesti ajastetun cronjob-tehtävän, joka korjaa MySQL-tietokannan valinnan mukaiselta ajanjaksolta. Mitä pidempi tuo aika on, sitä pidempään Sguil on poissa käytöstä. Tähänkin valittiin oletusarvon mukainen 7 päivää.

Sääntökokoelmaksi valittiin Talos + ET noGPL eli Snortin Community- ja Registered-paketit sekä Emerging Threadsin Open-kokoelma. Snortin sääntökokoelmia varten luotiin käyttäjätunnus Snortin sivuille, koska sääntöjen käyttö vaati ns. Oinkcoden. Sääntöpohjaiseksi NIDS:ksi valittiin Snort. Valintaan päädyttiin aiemman testauksen perusteella sekä sääntöjen paremman yhteensopivuuden vuoksi.

Kaikki verkonvalvontasensorit otettiin käyttöön. Snortin prosessien kuormituksen tasaamista varten käytettävän PF_RING:n minimiksi asetetiin oletusarvo 4096. Tämä arvo on pakettien lukumäärä, joka PF_RING-moduulin pitäisi vähimmillään

kyetä lisätä jonoon (Ntop n.d.). PF_RING:n avulla Security Onion pystyy suorittamaan useampia Snort-prosesseja samanaikaisesti (Security Onion Solutions_n n.d). Suricata ja Zeek käyttävät PF_RING:n sijaan nykyisin AF-PACKET-modulia, johon ollaan myöhemmin myös Snortin osalta siirtymässä sen 3.0-version julkaisun jälkeen (Security Onion Solutions_o n.d).

Seuraavaksi asennuksessa määritetään verkon valvonnan asetukset, kuten valvottavat verkkoliitännät, onko IDS:t käytössä, asetetaan samanaikaisesti ajettavien IDS-prosessien lukumäärä sekä sallitaanko järjestelmälle täysi pakettien kaappaus.

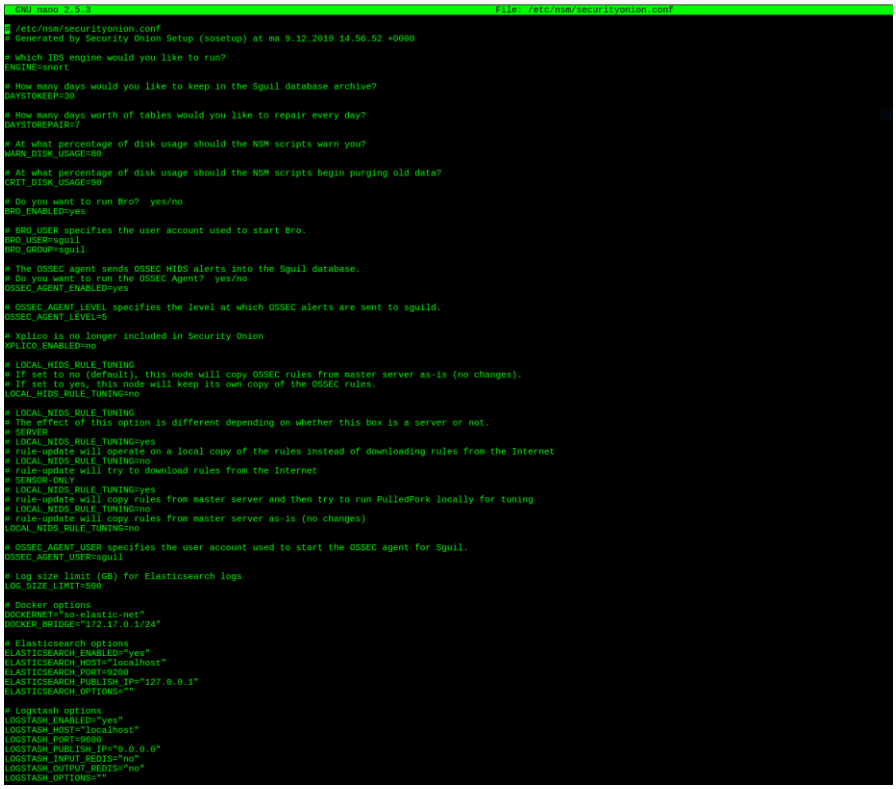
Koska tässä asennuksessa ei ole erillisiä sensoripalvelimia, vaan kaikki palvelut ovat käytössä yhdellä koneella IDS:t otettiin käyttöön kuten myös täysi pakettien kaappaus. Palvelimella jolle Security Onion asnnettiin, on 16-ytiminen prosessori, joten sekä Snort- että Zeek-prosesseja ajetaan kymmentä samanaikaisesti. Zeekin myös sallittiin erotella liikenteestä tiedostoja.

Järjestelmälle sallittiin täysi pakettien kaappaus ja tallennettavien PCAP-tiedostojen kooksi 150 megatavua. Netsniff-ng:lle sallittiin mmap I/O:n käyttö, joka tehostaa pakettien kaappausta, mutta vaatii enemmän ramia. Jokaista monitoroitavaa verkkoliitäntää varten varattiin 1 gigatavu muistia oletusarvon olleessa 64 megatavua. Järjestelmä alkaa tyhjätä lokeja kun 90 prosenttia tallennustilasta täyttyy.

Kotiverkoksi asetettiin myös oletusarvona olevat aliverkot 192.168.0.0/16, 10.0.0.0/8 ja 172.16.0.0/12, eli yksityiset IP-osoitteet. Muualta kuin näiden verkkojen IP-osoitteista tuleva liikenne katsotaan ulkoisiksi verkoiksi.

ElasticStack asetettiin käyttöön ja lokien kokorajaksi asetettiin 150 megatavua. Security Onion mahdollistaa lokien tallennuksen paikallisesti tai verkkokohteeseen. Tässä vaiheessa lokit asetettiin tallentumaan paikallisesti, mutta jatkossa ne voi olla perusteltua tallentaa verkkolevylle.

Valtaosa näistä määrityksistä on Security Onion pääasiallisessa asetustiedostossa, joka löytyy polusta /etc/nsm/securityonion.conf (kuvio 15). Osalle on oma asetustiedostonsa



KUVIO 15. securityonion.conf -tiedosto

Kun asennus on valmis, voi varmistaa palveluiden toiminnan komennolla:

```
sudo so-status
```

Mikäli jotkin palvelut eivät ole käynnistyneet oikein, ne voi käynnistää komennoilla:

```
sudo so-start
```

tai

```
sudo so-restart
```

Pakettien kaappauksen ja IDS-järjestelmien toimivuuden voi varmistaa syöttämällä terminaalissa komennon:

```
curl http://testmyids.com
```

6.4 Verkko

WPK-verkko käyttää runkonaan Ciscon kytkimiä ja reitittimiä, joten yksinkertaisin tapa ohjata verkon liikenne Security Onionille oli port mirroring, eli kytkimen portin kautta kulkevan liikenteen peilaaminen toiseen porttiin. Cisco käyttää port mirroringin toteutuksestaan nimitystä Switched Port Analyzer, eli SPAN.
SPANin heikkous kuitenkin on, ettei aivan kaikkia paketteja kyetä kaikissa tilanteissa peilaamaan. Tämä johtuu kahdesta syystä. SPAN ei ole kytkimen ensisijainen tehtävä, joten kytkimen kapasiteetin ylittyessä SPAN-prosessi ohitetaan, eikä paketteja peilata. Myös korruptoituneet ja liian pienet paketit sivuutetaan ja jätetään peilaamatta. (O'Neill 2018).
Nämä ongelmat voisi välttää käyttämällä TAP-laitetta (Terminal Access Point). TAP on passiivinen, joten se ei muokkaa paketin dataa, eikä tiputa paketteja. Sillä ei myöskään ole IP- tai MAC-osoitetta eikä se ole hakkeroitavissa. Sellaista ei kuitenkaan ollut käytettävissä tässä toteutuksessa. (Profitap 2019.)

WPK-verkko toimii TAMK:n verkon sisällä, ja näiden verkkojen rajalla on reititin, jonka kautta kaikki liikenne kulkee WPK-verkosta sisään ja ulos. Tämän reitittimen ja verkon väliin lisättiin kytkin, jonka kautta kulkeva liikenne peilataan Security Onion -palvelimelle. Näin saadaan mahdollisimman suuri määrä liikenteestä ohjattua Security Onionille tutkittavaksi.

SPAN- asetetaan kytkimelle komennoin:

```
Switch(config)# monitor session 1 source interface fastEthernet0/1
Switch(config)# monitor session 1 destination interface fastEthernet0/10 encapsulation dot1q
Switch(config)# end
```

(Cisco_d n.d.)

Tällöin portin fastEthernet0/1 sekä sisään että ulospäin suuntautuva liikenne peilataan porttiin fastEthernet0/10. Tämä portti kytkettiin Security Onionin sniffing interfaceen.

6.5 Asennuksen jälkeen

Koska Security Onion-palvelin sijaitsee fyysisesti WPK-verkon konesalissa, sille asennettiin etähallintaa varten Xrdp-ohjelmisto, joka mahdollistaa sen etäkäytön Microsoftin RDP-protokollaa käyttäen (Sorg n.d.).
Oletusasetuksillaan Security Onion sulkee kaikki UFW-palomuurin portit, estäen myös RDP:n liikenteen, joten etäkäyttöä varten pitää ne avata. Lisäksi toimiakseen xrdp vaatii sen kanssa yhteensopivan työpöytäympäristön ja asettaa xrdp käyttämään sitä. Security Onionin käyttämä Gnome soveltuu käytettäväksi xrdp:n kanssa, mutta sen voi halutessaan korvata myös esim. Xfce4:llä (Secatlas 2015).
Xrdp:n asennukseen käytettiin Security Onionin kehityksessä toimivan Bryant Treaclen luomaa skriptiä, joka asentaa tarvittavat tiedostot, sekä avaa RDP:n vaatimat portit. Skriptiä suorittaessa voi asettaa käyttäjän jolle etäyhteyden käyttö sallitaan, sekä IP-osoitteen tai aliverkon, josta yhteyttä sallitaan käyttää. (Treacle 2018.)

6.6 Asetukset

Perusasetuksillaan Security Onion tuottaa paljon ns. false positive -hälytyksiä, eli hälytyksiä, jotka eivät todellisuudessa ole oikeita tietoturvauhkia. Siksi sen asetuksista on syytä jättää tietyt säännöt huomioimatta tai rajata niistä aiheutuvia hälytyksiä. Tämä tehdään muokkaamalla Security Onionin asetustiedostoja.
Koska toimeksiantajan toiveen mukaisesti Security Onionin on tarkoitus tuottaa mahdollisimman paljon hälytyksiä tarkasteltavaksi, false positiveja ei kuitenkaan karsittu. Niitä kuitenkin tarkisteltiin toteutuksen yhteydessä järjestelmään tutustuessa.

WPK:n kaltaisessa verkossa, jossa valtaosassa käytettävistä koneista on Windows 10-käyttöjärjestelmä, esimerkiksi USB-laitteiden metadatan lähettäminen Microsoftille tuottaa suuren määrän turhia hälytyksiä, ja ne voisi jättää huomioimatta. Sama koskee Windowsin päivityksiä. Myös kaikki tiedostonjako- ja pilvitallennuspalvelut kuten Dropbox tai Google Drive, joita monet opiskelijat verkon

koneilla käyttävät aiheuttavat hälytyksiä. Näiden kaltaiset false positiveja aiheuttavat säännöt voisi lähes riskittömästi poistaa käytöstä kokonaan. Tämä voidaan tehdä lisäämällä säännön signature ID, eli sid asetustiedostoon /etc/nsm/pulledpork/disablesid.conf.

Osaa hälytyksistä, kuten TCP-protokollan kättelyyn liittyviä hälytyksiä, ei useinkaan kannata poistaa kokonaan käytöstä, sillä ne voivat viitata myös oikeaan hyökkäykseen. Tällöin hälytysten määrää ja tiheyttä kannattaa muokata. Tämä tapahtuu /etc/nsm/rules/threshold.conf -tiedostoa muokkaamalla. Hälytyksille voi asettaa kolmen tyyppisiä suodattimia. Hälytys voidaan rajoittaa määrällisesti, jolloin vain valitun mukainen määrä hälytyksiä luodaan annetun aikarajan sisällä. Määrän täytyttyä tapahtumat sivuutetaan. Hälytyksille voidaan asettaa kynnys, jolloin valitaan, kuinka monen samanlaisen tapahtuman välein siitä aiheutuu hälytys. Lisäksi voidaan asettaa suodatin näiden yhdistelmänä. (Security Onion Solutions_p n.d.)

6.7 Analysointivirtuaalikoneen luominen

Lopuksi verkkoon asennettiin analysointia varten Windows-palvelimelle Security Onionin ohjeistuksen mukaan analysointi-virtuaalikone, jolle ei otettu käyttöön paketinkaappausta tai tunkeutumisen havaitsemisjärjestelmiä. Kaikki analysointityökalut kuten Wireshark kuuluvat kuitenkin perusasennukseen, joten ne ovat ylläpidon käytettävissä. Lisäksi koneelle luotiin oikopolut Sguilin, Kibanan, Cyberchefin ja Squertin käyttöön, niin että ne avaavat automaattisesti yhteyden Security Onion -palvelimelle.
Kuten Security Onionin pääasennuksessa, myös virtuaalikoneelle asennettiin Xrdp etäkäyttöä varten. Xrdp:llä käyttäessä kuitenkin ikkunointijärjestelmä on rajoittuneempi. Sen vuoksi virtuaalikoneesta luotiin myös valmis levykuva, jonka voi tarvittaessa asentaa ylläpitäjän omalle koneelle, jolloin sitä ei tarvitse käyttää etäyhteyden yli. Vaihtoehtoisesti konetta voi käyttää ottamalla etäyhteyden palvelimelle, jolle se on asennettu, ja avata se siltä.

7 POHDINTA

Opinnäytetyön aihe on hyvin ajankohtainen, koska sekä yritykset, organisaatiot että yksityiset ihmiset ovat enenevässä määrin riippuvaisia tietojärjestelmistä. Siksi niiden suojaaminen on jatkuvasti tärkeämpää yhteiskunnan toiminnan kannalta.

Työn tavoitteena oli parantaa WPK-verkon havainnointikykyä hyökkäysten ja tietomurtojen varalta käyttäen Security Onion -verkonvalvonta-alustaa. Security Onion on hyvin mielenkiintoinen työn kohteena, koska se on ilmainen ja käytännössä täysin vapaata koodia, mutta tarjoaa erittäin kattavan kokoelman erilaisia työkaluja verkon valvontaan. Security Onionia on tarkoitus hyödyntää myös opetuskäytössä. Tämä kuitenkin aiheutti hieman ristiriitaisen tilanteen, kun samaan aikaan oli tarve saada suuri määrä hälytyksiä järjestelmästä, mutta järjestelmän tietoturvallisen toiminnan kannalta olisi ollut tarpeen rajata hälytyksien määrää.

Myös Security Onionin avoimuus aiheutti hieman ongelmia. Kaupallisten ohjelmistojen tuki on huomattavasti laajempaa kuin avoimen lähdekoodin ohjelmistojen. Joissain tilanteissa tietoa ei ollut löydettävissä juuri lainkaan ja toisissa tiettyjä asioita joutui etsimään valtavasta informaatiotulvasta erilaisten keskustelupalstojen, blogien, ohjeiden yms. seasta. Lisäksi koska Security Onion alkuaan yhden henkilön alulle pistämä projekti, sen kokoonpano muuttuu hyvin nopeasti ja siksi tietojen ajantasaisuus voi aiheuttaa hankaluuksia, kun työkalut vaihtuvat toisiin.

Työn lopputuloksena on kuitenkin saavutettu toimiva järjestelmä, joka toimii itsenäisesti ja mahdollistaa varsin tarkan ja yksityiskohtaisen verkon valvonnan WPK-verkon ylläpidon käyttöön. Sitä on mahdollista myös käyttää etäyhteydellä, joten käyttäjä ei ole sidottu aikaan tai paikkaan. Lisäksi se tarjoaa mahdollisuuden hyödyntää sitä opetuskäytössä.

Jatkokehityksen kannalta järjestelmän käytön tasapainoa olisi hyvä tarkentaa hälytysten määrän ja suodatuksen kannalta, sekä harkita työkalujen maksullisten versioiden hyödyntämistä.

LÄHTEET

Ashoor & Gore. 2011. Importance of Intrusion Detection System (IDS). https://pdfs.semanticscholar.org/2a84/3c5894f2ef76c1d1b5d9829c0cc88852b4e5.pdf

Axelsson. 2000. Taxonomy and Survey of Collaborative Intrusion Detection. http://neuro.bstu.by/ai/To-dom/My_research/Paper-0-again/For-research/D-mining/Anomaly-D/Intrusion-detection/taxonomy.pdf

Baker., Kohlenberg, Beale & Esler. 2007. Snott IDS and IPS Toolkit

Bejtlich R. 2013. The Practice of Network Security Monitoring

Berman D. 2019. The Complete Guide to the ELK Stack https://logz.io/learn/complete-guide-elk-stack/

Burks D. 2018. Security Onion Blog. Wazuh 3.6.1, Elastic 6.4.1, and associated components are now available for Security Onion 16.04!. https://blog.securityonion.net/2018/10/wazuh-361-elastic-641-and-associated.html

Burks D. Security Onion Blog - Squert development. Luettu 27.02.2020 https://blog.securityonion.net/2016/09/squert-development.html

CyberChef Readme. n.d. Luettu 30.3.2020 https://github.com/gchq/CyberChef/blob/master/README.md

Carr J. 2007. Snort: Open Source Network Intrusion Prevention https://www.esecurityplanet.com/network-security/Snort-Open-Source-Network-Intrusion-Prevention-3681296.htm

Cisco_a. n.d. Snort FAQ - How is the Snort software licensed?. Luettu 19.10.2019 https://www.snort.org/faq/how-is-the-snort-software-licensed

Cisco_b. n.d. Snort FAQ - What is Snort. Luettu 20.10.2019 https://snort.org/faq/what-is-snort

Cisco_c. n.d. Snort FAQ - What are the differences in the rule sets? Luettu 20.10.2019 https://snort.org/faq/what-are-the-differences-in-the-rule-sets

Cisco_d. n.d. Catalyst Switched Port Analyzer (SPAN) Configuration Example. Luettu 5.1.2020 https://www.cisco.com/c/en/us/support/docs/switches/catalyst-6500-series-switches/10570-41.html.

GCHQ. 2016. CyberChef - the Cyber "Swiss Army Knife". https://www.gchq.gov.uk/news/cyberchef-cyber-swiss-army-knife

Elastic_a. Logstash Introduction. Luettu 25.10.2019. https://www.elastic.co/guide/en/logstash/current/introduction.html

Elastic_b. Elasticsearch introduction. Luettu 25.10.2019. https://www.elastic.co/guide/en/elasticsearch/reference/current/elasticsearch-intro.html

Elastic_c. Kibana Introduction. Luettu 26.10.2019 https://www.elastic.co/guide/en/kibana/current/introduction.html

Elastic_d. Stashing Your First Event. Luettu 1.4.2020 https://www.elastic.co/guide/en/logstash/current/first-event.html

Elson D. 2000. Intrusion Detection, Theory and Practice https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=0d6b1119-041a-417a-9974-7f9037938328&

Gupta. R & Gupta Y. 2017. Mastering Elastic Stack

Gupta, Singha, Verma, & Singh. 2018. Intrusion Detection Prevention System using SNORT. https://www.irjet.net/archives/V4/i4/IRJET-V4I4439.pdf

Hein D. 2019. Why Is Network Monitoring Important for Enterprises? https://solutionsreview.com/network-monitoring/why-is-network-monitoring-important-for-enterprises/

Hjelmvik E. n.d. Passive Network Security Analysis with NetworkMiner. Luettu 2.3.2020. https://www.forensicfocus.com/passive-network-security-analysis-networkminer

Kent and Mell. 2007. Guide to Intrusion Detection and Prevention Systems (IDPS). https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=50951

MHC Datacomm Inc. 2019. Cyber Crime and the Importance of Intrusion Detection Systems. https://mhcdce.com/news/cyber-crime-and-the-importance-of-intrusion-detection-systems/

Morrow S. 2019. Peeling the Onion - Security Onion OS. https://resources.infosecinstitute.com/peeling-the-onion-security-onion-os/

Mäntylahti P. 2003. IDS-järjestelmät. https://www.tivi.fi/uutiset/ids-jarjestelmat/50eb3822-855b-3ac9-a971-a64a2b35de7d

Netresec. n.d. NetworkMiner. Luettu 30.2.2020. https://www.netresec.com/?page=networkminer

Netsniff-ng. n.d. Luettu 7.2.2020 http://netsniff-ng.org/

Northcutt S. & Novak J. 2002. Network Intrusion Detection, Third Edition

Ntop. PF_RING Documentation. Luettu 27.2.2020 https://www.ntop.org/guides/pf_ring/get_started/git_installation.html

O'neill. 2018. "SPAN Port Or Tap? CSO Beware. https://www.profitap.com/wp-content/uploads/SPAN-Port-or-TAP-CSO-Beware-Tim-O-Neill.pdf

Open Information Security Foundation_a. n.d. Suricata wiki - What is Suricata. Luettu 10.11.2019 https://redmine.openinfosecfoundation.org/projects/suricata/wiki/What_is_Suricata

Open Information Security Foundation_b. n.d. Suricata Documentation - What is Suricata. Luettu 11.11.2019 https://suricata.readthedocs.io/en/suricata-5.0.0/what-is-suricata.html

Open Information Security Foundation_c. n.d. Suricata Documentation - GNU General Public License. Luettu 11.11.2019 https://suricata.readthedocs.io/en/suricata-5.0.0/licenses/gnu-gpl-v2.0.html

Profitap. 2019. Tap Vs SPAN. https://www.profitap.com/wp-content/uploads/TAP-vs-SPAN.pdf

Rao L. 2013. Cisco Acquires Cybersecurity Company Sourcefire For $2.7B. https://techcrunch.com/2013/07/23/cisco-acquires-cybersecurity-company-sourcefire-for-2-7b/

Roesch. 1999. Snort – Lightweight Intrusion Detection for Networks. https://www.usenix.org/legacy/event/lisa99/full_papers/roesch/roesch.pdf

Sanders C. & Smith J. 2013. Applied Network Security Monitoring

Sarmah. 2001. Intrusion Detection Systems:Definition, Need and Challenges. https://www.sans.org/reading-room/whitepapers/detection/intrusion-detection-systems-definition-challenges-343

Secatalas. 2015. Security Onion and XRDP. https://secatlas.wordpress.com/2015/03/09/security-onion-and-xrdp/.

SecTools. n.d. NetworkMiner. Luettu 2.3.2020. https://sectools.org/tool/networkminer/

SecTools. n.d. Top 125 Network Security Tools. Luettu 1.3.2020. https://sectools.org/tag/sniffers/

Security Onion Solutions_a. n.d. Security Onion documentation – Introduction Luettu 17.4.2020 https://securityonion.readthedocs.io/en/latest/introduction.html

Security Onion Solutions_b. n.d. Security Onion documentation – About. Luettu 7.5.2020 https://securityonion.readthedocs.io/en/latest/about.html

Security Onion Solutions_c. n.d. Security Onion documentation – Architecture. Luettu 1.5.2020 https://securityonion.readthedocs.io/en/latest/architecture.html

Security Onion Solutions_d. n.d. Security Onion documentation – Managing Rules. Luettu 25.2.2020 https://securityonion.readthedocs.io/en/latest/rules.html

Security Onion Solutions_e. Security Onion documentation – Zeek. Luettu 25.2.2020 https://securityonion.readthedocs.io/en/latest/zeek.html

Security Onion Solutions_f. n.d. Security Onion documentation – Wazuh. Luettu 25.2.2020 https://securityonion.readthedocs.io/en/latest/wazuh.html

Security Onion Solutions_g. Security Onion documentation – Sguil. Luettu 12.10.2019 https://securityonion.readthedocs.io/en/latest/sguil.html

Security Onion Solutions_h. Security Onion documentation – CapME. Luettu 22.2.2020 https://securityonion.readthedocs.io/en/latest/capme.html

Security Onion Solutions_i. n.d. Security Onion documentation – Squert. Luettu 28.2.2020 https://securityonion.readthedocs.io/en/latest/squert.html

Security Onion Solutions_j. n.d. Security Onion - Wireshark. Luettu 25.2.2020 https://securityonion.readthedocs.io/en/latest/wireshark.html

Security Onion Solutions_k. n.d.Security Onion - NETWorkMiner. Luettu 30.2.2020 https://securityonion.readthedocs.io/en/latest/networkminer.html

Security Onion Solutions_l. Security Onion documentation – Connecting to Sguild. Luettu 5.1.2020 https://securityonion.readthedocs.io/en/latest/connecting-to-sguil.html

Security Onion Solutions_m. n.d. Security Onion documentation – Analyst VM. Luettu 9.10.2019 https://securityonion.readthedocs.io/en/latest/analyst-vm.html

Security Onion Solutions_n. n.d. Security Onion documentation – PF-RING. Luettu 27.2.2020 https://securityonion.readthedocs.io/en/latest/pf-ring.html

Security Onion Solutions_o. n.d.Security Onion documentation – AF-PACKET. Luettu 27.2.2020 https://securityonion.readthedocs.io/en/latest/af-packet.html

Security Onion Solutions_p. n.d. Security Onion documentation – Managing Alerts. Luettu 28.2.2020 https://securityonion.readthedocs.io/en/latest/alerts.html

Smith J. 2013. Applied Network Security Monitoring

Sorg J. n.d. Xrdp Overview. Luettu 5.12.2019 http://xrdp.org/

Techopedia. 2011. Network-Based Intrusion Detection System (NIDS). Luettu https://www.techopedia.com/definition/12941/network-based-intrusion-detection-system-nids.

Techopedia. 2017. Host-Based Intrusion Detection System (HIDS). Luettu https://www.techopedia.com/definition/12826/host-based-intrusion-detection-system-hids.

Thomas T. 2005. Verkkojen tietoturva: perusteet

Treacle B. 2018. Security_Onion_XRDP_Setup. https://github.com/bryant-treacle/Security_Onion_XRDP_Setup

Vaggalis N. 2019. CyberChef - The Developer's Ultimate Toolbox https://www.i-programmer.info/news/90-tools/13117-cyberchef-the-developers-ultimate-toolbox.html

Visscher B. 2014. Sguil: The Analyst Console for Network Security Monitoring http://bammv.github.io/sguil/index.html

Wazuh Inc_a. Wazuh documentation -Welcome to Wazuh. Luettu 26.2.2020 https://documentation.wazuh.com/3.10/index.html

Wazuh Inc_b. Wazuh documentation - Migrating from OSSEC. Luettu 26.2.2020 https://documentation.wazuh.com/3.10/migrating-from-ossec/index.html

Wireshark FAQ. n.d. Luettu 29.02.2020. https://www.wireshark.org/faq.html

Wireshark. n.d. About Wireshark. Luettu 29.02.2020. https://www.wireshark.org/

Zeek. n.d. Luettu 25.2.2020 https://www.zeek.org/